



UNIVERSIDADE ESTADUAL DE CAMPINAS

Instituto de Matemática, Estatística
e Computação Científica

Adriano Francisco Monteiro dos Santos

MODELAGEM LAGRANGIANA E SIMULAÇÕES DE SISTEMAS
MECÂNICOS CONSTITUÍDOS POR MASSA-MOLA E PÊNDULOS

CAMPINAS
2018

Adriano Francisco Monteiro dos Santos

**MODELAGEM LAGRANGIANA E SIMULAÇÕES DE SISTEMAS
MECÂNICOS CONSTITUÍDOS POR MASSA-MOLA E PÊNDULOS**

Dissertação apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Matemática Aplicada e Computacional.

Orientador: Simão Nicolau Stelmastchuk

Este exemplar corresponde à versão final da dissertação defendida pelo aluno Adriano Francisco Monteiro dos Santos e orientada pelo Prof. Dr. Simão Nicolau Stelmastchuk.

**CAMPINAS
2018**

Agência(s) de fomento e nº(s) de processo(s): Não se aplica.

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Silvania Renata de Jesus Ribeiro - CRB 8/6592

Santos, Adriano Francisco Monteiro dos, 1982-
Sa59m Modelagem lagrangiana e simulações de sistemas mecânicos constituídos por massa-mola e pêndulos / Adriano Francisco Monteiro dos Santos. – Campinas, SP : [s.n.], 2018.

Orientador: Simão Nicolau Stelmastchuk.
Dissertação (mestrado profissional) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Euler-Lagrange, Equações de. 2. Simulação computacional. 3. Pêndulo.
I. Stelmastchuk, Simão Nicolau, 1977-. II. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Lagrangian modeling and simulations of mechanical systems constituted by spring-mass and pendulums

Palavras-chave em inglês:

Euler-Lagrange equations

Computer simulation

Pendulum

Área de concentração: Matemática Aplicada e Computacional

Titulação: Mestre em Matemática Aplicada e Computacional

Banca examinadora:

Simão Nicolau Stelmastchuk [Orientador]

Jayme Vaz Júnior

Fabiano Borges da Silva

Data de defesa: 23-11-2018

Programa de Pós-Graduação: Matemática Aplicada e Computacional

Dissertação de Mestrado Profissional defendida em 23 de novembro de 2018 e aprovada pela banca examinadora composta pelos Profs. Drs.

Prof(a). Dr(a). SIMÃO NICOLAU STELMASTCHUK

Prof(a). Dr(a). JAYME VAZ JÚNIOR

Prof(a). Dr(a). FABIANO BORGES DA SILVA

A Ata da Defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria de Pós-Graduação do Instituto de Matemática, Estatística e Computação Científica.

Aos meus filhos João Elias e George.

Agradecimentos

Primeiramente à Deus por ter me dado saúde e força.

Aos meus pais Francisco e Raquel e irmãos Luciano e Júnior pelo apoio e incentivo que me deram.

A minha esposa Adriana, pelo amor, companheirismo, por estar comigo em todos os momentos dando força e otimismo.

Ao orientador, Prof. Dr. Simão Nicolau Stelmastchuk, por ter aceitado me orientar, pela paciência e por depositar sua confiança em mim durante o trabalho.

Aos meus amigos que fiz durante o Mestrado, em especial Émerson, Juscimar e George, que em todos os módulos estiveram comigo.

Aos meus amigos do IFMA campus Imperatriz, em especial aos professores Bosco e Remi que me apoiaram nessa jornada.

Aos professores do mestrado pelo conhecimento transmitido.

Aos professores da Banca examinadora.

Resumo

A modelagem lagrangiana fornece um procedimento para obter um sistema de equações diferenciais de segunda ordem que descreve o movimento de um sistema de partículas. O presente trabalho realiza simulações de alguns modelos bidimensionais e tridimensionais de sistemas constituídos por massa-mola e pêndulos amortecidos pela função de dissipação de Rayleigh. Nós usamos o software Mathematica para calcular soluções dos sistemas de equações obtidos pela modelagem lagrangiana e fazer simulações dos modelos estudados. Os passos para implementar as simulações são detalhados e servem de base para outros modelos.

Palavras-chave: Equações de Euler-Lagrange, Simulação computacional, Pêndulo.

Abstract

Lagrangian modeling provides a procedure for obtaining a system of second order differential equations describing the motion of a particle system. The present work performs simulations of some two-dimensional and three-dimensional models consisting of spring-mass systems and pendulums damped by the Rayleigh dissipation function. We use the Mathematica software to obtain solutions of the system of equations for Lagrangian modeling and to build simulations of the models studied. The steps to implement the simulations are detailed and serve as the basis for other models.

Keywords: Euler-Lagrange equations, Computer simulation, Pendulum.

Lista de Ilustrações

1.1	Movimento harmônico simples	19
1.2	Pêndulo planar	19
1.3	Pêndulo duplo	21
1.4	Rolamento sem deslizamento	22
1.5	Deslocamento virtual	22
1.6	Problema da braquistócrona	32
2.1	Função θ resultado de uma solução numérica do pêndulo planar.	36
2.2	Gráfico da função θ solução do pêndulo planar	36
2.3	Trajetoória do pelo pêndulo planar	37
2.4	Gráficos da função θ e trajetória do pêndulo planar com dissipação	38
2.5	Animação do traçado do pêndulo	39
2.6	Animação do modelo do pêndulo planar	40
2.7	Janelas de simulações de pêndulos planares	41
2.8	Simulação do oscilador harmônico simples	42
2.9	Simulação do oscilador harmônico com dissipação	43
2.10	Pêndulo planar com haste elástica	44
2.11	Trajetoória do pêndulo planar com haste elástica	46
2.12	Simulação do pêndulo planar com haste elástica	46
2.13	Trajetoória do pêndulo planar com haste elástica e dissipação	47
2.14	Simulação do pêndulo planar com haste elástica e dissipação	47
3.1	Pêndulo esférico	48
3.2	Trajetoória do pêndulo esférico	50
3.3	Simulação do pêndulo esférico	50
3.4	Trajetoória do pêndulo esférico com dissipação	51
3.5	Simulação do pêndulo esférico com dissipação	52
3.6	Pêndulo esférico com haste elástica	52
3.7	Trajetoória do pêndulo esférico com haste elástica	53
3.8	Simulação do pêndulo esférico com haste elástica	54
3.9	Trajetoória do pêndulo esférico com haste elástica e dissipação	55
3.10	Simulação do pêndulo esférico com haste elástica e dissipação	55

Lista de Algoritmos

2.1	Estudos de sistema de equações de movimento	35
2.2	Simulação interativa.	40

Lista de Códigos-fonte

2.1	Solução numérica do pendulo planar	36
2.2	Plot do gráfico da função θ	36
2.3	Trajectoria do pêndulo planar	37
2.4	Animação do traçado do pêndulo	38
2.5	Animação do modelo do pêndulo planar	39
2.6	Trajectoria do pêndulo planar com haste elástica	45
3.1	Trajectoria do pêndulo esférico	49
A.1	NDSolve - Pêndulo planar	58
A.2	NDSolve: Pêndulo planar com dissipação	58
A.3	Plot: Solução das equações de movimento planares com ou sem dissipação	59
A.4	ParametricPlot: Trajetória dos pêndulos planares com ou sem dissipação	59
A.5	Comando Manipulate	59
A.6	Pêndulo plano	59
A.7	Pêndulo plano com dissipação	60
A.8	Oscilação elástica	62
A.9	Oscilação elástica com dissipação	63
A.10	Trajectoria pêndulo com haste elástica	64
A.11	Pêndulo plano com haste elástica	65
A.12	Trajectoria pêndulo com haste elástica e dissipação	66
A.13	Pêndulo plano com haste elástica e dissipação	67
A.14	Trajectoria pêndulo esférico	68
A.15	Pêndulo esférico	69
A.16	Trajectoria pêndulo esférico com dissipação	71
A.17	Pêndulo esférico com dissipação	72
A.18	Trajectoria pêndulo esférico com haste elástica	73
A.19	Pêndulo esférico com haste elástica	74
A.20	Trajectoria pêndulo esférico com haste elástica e dissipação	76
A.21	Pêndulo esférico com haste elástica e dissipação	77

Sumário

Introdução	13
1 Equação de Euler-Lagrange	14
1.1 Contexto Histórico	14
1.2 Modelagem newtoniana	14
1.2.1 Teoremas de conservação	16
1.2.2 Equações de movimento para uma partícula	18
1.3 Modelagem lagrangiana	20
1.3.1 Vínculos	20
1.3.2 Deslocamentos virtuais	22
1.3.3 Trabalhos virtuais	23
1.3.4 Coordenadas generalizadas	24
1.3.5 Equação de Euler-Lagrange	26
1.3.6 Potenciais generalizados	29
1.3.7 Função de dissipação de Rayleigh	30
1.4 Cálculo das variações	31
2 Pêndulos planares	34
2.1 Modelagem com o Mathematica	34
2.1.1 Soluções numéricas	34
2.1.2 Simulações	38
2.2 Problemas bidimensionais	41
2.2.1 Modelagem lagrangiana do oscilador harmônico simples	41
2.2.2 Pêndulo planar com haste elástica	43
3 Pêndulos esféricos	48
3.1 Pendulo esférico	48
3.1.1 Pendulo esférico com dissipação	51
3.1.2 Pêndulo esférico com haste elástica	52
3.1.3 Pêndulo esférico com haste elástica e dissipação	54
Referências	57
Apêndice A Projeto Computacional	58

Introdução

A modelagem de movimentos de sistemas de partículas por meio de equações, onde é possível determinar em função do tempo a posição de cada partícula conhecendo as forças que atuam sobre elas e suas posições iniciais, são estudadas pela Mecânica Clássica. Newton formou a base dessa ciência com suas leis de movimento, mas à medida que a complexidade do sistema aumenta, a modelagem newtoniana perde sua eficácia pois ela compreende todas as forças de um sistema. A modelagem lagrangiana, de forma conveniente, separa as forças existentes em dois tipos: as forças aplicadas, que são consideradas as verdadeiras causas de um movimento, e as forças de vínculo, que são forças que garantem a geometria e a cinemática do problema. Dessa forma, a modelagem lagrangiana reduz a quantidade de forças necessárias para modelar um sistema em relação à abordagem newtoniana.

Com as equações de Euler-Lagrange temos uma ferramenta que facilita encontrar as equações de movimento de um problema, porém, elas fornecem um sistema de equações diferenciais de segunda ordem que, muitas vezes, são resolvidas apenas por algoritmos de aproximação numérica, pois exigiriam muito esforço para resolver sem auxílio computacional. Com objetivo de desenvolver mais efetivamente a modelagem de Sistemas Mecânicos, construímos simulações de modelos de pêndulos planares e esféricos e para isso utilizamos o software Wolfram Mathematica para resolver e desenhar um conjunto de gráficos de soluções, formando simulações de problemas onde é possível alterar interativamente as condições iniciais e observar e responder questões sobre o efeito produzido pela mudança de cada parâmetro.

O desenvolvimento deste trabalho está organizado em três capítulos: o capítulo 1 apresenta as modelagens newtoniana e lagrangiana, mostra como obter as equações de Euler-Lagrange e uma forma de calcular amortecimento de um sistema mecânico com a função de dissipação de Rayleigh. O capítulo 2 apresenta a modelagem e simulação de problemas de pêndulos bidimensionais. O capítulo 3 apresenta a modelagem e simulação de pêndulos tridimensionais.

1 Equação de Euler-Lagrange

Neste primeiro capítulo apresentaremos o desenvolvimento das equações de Euler-Lagrange a partir da Mecânica Newtoniana e dos deslocamentos virtuais a fim de obter equações de movimento para sistemas partículas. Para este capítulo foram usados como referências [1], [2], [3], [4] e [5].

1.1 Contexto Histórico

Em 1696, Johann Bernoulli publicou um problema desafiando os matemáticos de sua época a encontrar a trajetória em um plano vertical de uma partícula que se move de um ponto A, sob influência de próprio peso, a outro ponto B no menor tempo possível, de forma que os pontos não estejam sob uma reta horizontal nem vertical. No ano seguinte, Bernoulli publicou seis soluções submetidas, foram elas: a sua própria solução, a do seu irmão Jakob Bernoulli, de Newton, de Leibniz, de Tschirnhaus e do Marquês de L'Hopital.

Problemas de otimização não eram novidade mas esses eventos marcam o nascimento do cálculo das variações porque foram seguidos por um período de intensa atividade em problemas similares que levaram a maturidade do campo. Leonhard Euler, que foi aluno de Johann Bernoulli, publicou em 1744 o livro *The Method of Finding Plane Curves that Show Property of Maximum or Minimum* onde mostrou um procedimento para escrever o que ficou conhecido com equação de Euler. Lagrange se interessou por problemas variacionais e viu como chegar ao mesmo resultado de Euler reduzindo a um processo analítico sem a necessidade de usar percepções geométricas e em 1755 escreveu para Euler mostrando seus resultados. Hoje, as equações que são conhecidas como equações de Euler-Lagrange, são muito utilizadas para resolver problemas de mecânica, principalmente quando os problemas tornam-se muito complexos para serem resolvidos pela abordagem newtoniana. A Mecânica é um ramo da Física que estuda a dinâmica de partículas descrevendo matematicamente o movimento dos corpos.

1.2 Modelagem newtoniana

As leis fundamentais da mecânica foram fornecidas por Isaac Newton (1642-1727), suas três leis do movimento foram enunciadas em 1687 na sua principal obra **Philosophiae Naturalis Principia Mathematica**, em português **Principios matemáticos de filosofia natural**, elas formam a base da Mecânica Clássica. Uma breve discussão que permite prosseguir no estudo da Mecânica Clássica é feita a seguir com objetivo de apresentar conceitos necessários para o desenvolvimento das equações de Euler-Lagrange. O leitor deve estar ciente que as leis de Newton não podem ser demonstradas pois são baseadas em fatos experimentais. Os enunciados modernos assim como encontrado em [1] são:

- I. *Existem sistemas de referências, chamados de inerciais, em que um corpo permanece em repouso ou movimento uniforme, exceto sob atuação de uma força.*

A primeira lei, também conhecida como *Lei da Inércia*, define **sistemas de referencial inercial** onde a segunda lei pode ser aplicada. Um sistema de referência pode ser explicado como um espaço descrito por um sistema de coordenadas qualquer (cartesianas, polar, esférica, etc.) e um relógio que marca a passagem uniforme do tempo. Neles podemos anotar a posição de uma partícula do sistema a cada instante t e definir uma função para sua trajetória. Foi adotado nesse trabalho, por ser bastante usual, nomear esse tipo função de trajetória por \mathbf{r} , que se trata de uma função vetorial do tipo $\mathbf{r} = \vec{r}(t)$ e, para facilitar a leitura, o uso apenas do nome da função em negrito como notação para funções vetoriais dependente do tempo.

Dada uma função \mathbf{r} , a velocidade, denominada por \mathbf{v} , é definida como a direção e a taxa de variação da posição com o tempo

$$\mathbf{v} = \frac{d\mathbf{r}}{dt},$$

e a aceleração, denominada por \mathbf{a} , é definida como a taxa de variação da velocidade com o tempo

$$\mathbf{a} = \frac{d\mathbf{v}}{dt}.$$

Em textos na área da Física, é comum o uso da notação com um ou dois pontos acima do nome da função indicando que se trata, respectivamente, da derivada primeira ou derivada segunda em relação ao tempo. Assim, a velocidade pode ser escrita como $\dot{\mathbf{r}}$ e a aceleração pode ser indicada tanto como $\dot{\mathbf{v}}$ ou como $\ddot{\mathbf{r}}$.

- II. *Em um referencial inercial, um corpo sob a atuação de uma força se move de tal forma que a taxa temporal da variação de movimento é igual a força.*

A segunda lei é conhecida como *Lei Fundamental da Dinâmica* e somente é válida em um sistema de referencial inercial. A definição de Newton de **quantidade de movimento** (\mathbf{p}) é dada por

$$\mathbf{p} \equiv m\mathbf{v},$$

onde m é a massa e \mathbf{v} é a velocidade de uma partícula. Assim, a equação que expressa a segunda lei pode ser escrita como

$$\mathbf{F} = \frac{d\mathbf{p}}{dt} = \frac{d}{dt}(m\mathbf{v}). \quad (1.2.1)$$

O conceito de massa pode ser entendido, intuitivamente, como a quantidade de matéria de um corpo resultante de sua densidade e volume. No entanto, é uma ideia redundante, pois a densidade é definida em função da massa e volume. Uma definição mais elaborada, porém sem envolver conceitos modernos de relatividade, pode ser obtida em decorrência da terceira lei.

Dependendo da aplicação física, um problema pode envolver acréscimo, perda ou mesmo não ter variação de massa. São inúmeras, e possivelmente maioria, as aplicações que envolvem massa constante, por isso é muito comum desenvolver a equação (1.2.1) na forma

$$\mathbf{F} = m\left(\frac{d\mathbf{v}}{dt}\right) = m\dot{\mathbf{v}}$$

e por (1.2), chega-se a equação

$$\mathbf{F} = m\mathbf{a},$$

que é a forma mais conhecida do enunciado da Segunda Lei de Newton e essa será a equação utilizada no decorrer desse texto, pois as aplicações estudadas não envolvem variação de massa.

III. *Dois corpos exercem forças entre si, essas forças serão iguais em magnitude e opostas em termos de direção.*

Deve-se acrescentar que a validade dessa lei não é geral pois não pode ser aplicada para forças que dependam da velocidade dos corpos em interação como, por exemplo, são as forças entre cargas elétricas se movendo. A terceira lei é conhecida como *Lei da ação e Reação* e, como já mencionado, permite uma definição melhor para massa no contexto da Mecânica Clássica, como uma medida que determina a aceleração em um referencial inercial, evitando a redundância de definir a massa por meio da densidade.

Se \mathbf{F}_1 é a ação de uma força de uma partícula i sobre uma partícula j então existe \mathbf{F}_2 tal que

$$\mathbf{F}_1 = -\mathbf{F}_2$$

pela segunda lei segue que

$$\begin{aligned}\frac{d\mathbf{p}_1}{dt} &= -\frac{d\mathbf{p}_2}{dt} \\ m_1 \left(\frac{d\mathbf{v}_1}{dt} \right) &= -m_2 \left(\frac{d\mathbf{v}_2}{dt} \right) \\ \frac{m_2}{m_1} &= -\frac{\mathbf{a}_1}{\mathbf{a}_2}.\end{aligned}$$

Se m_1 for definida como uma massa unitária, a massa m_2 pode ser determinada experimentalmente por medições adequadas das acelerações usando instrumentos precisos apropriados. Uma forma bem comum de fazer essa medição é por meio da pesagem utilizando-se uma balança e comparando com um peso padrão, considerando que o peso é simplesmente a força gravitacional atuando sobre um corpo.

Outra consequência derivada dessa lei é que ela garante a conservação dos momentos linear e angular total de sistemas isolados. A próxima seção descreve em forma de teoremas essas consequências.

1.2.1 Teoremas de conservação

Serão enunciados teoremas de conservação da quantidade de momento linear, da quantidade de momento angular e da energia em campos conservativos, note que a demonstração não constitui uma prova, somente deriva as várias consequências das leis de Newton, ou seja, se essas leis são válidas em uma certa situação, então a quantidade de movimento e a energia serão conservados.

Teorema 1.2.1. *A quantidade de movimento linear total \mathbf{p} de uma partícula é conservada quando a força total sobre ela é zero.*

Se uma partícula não encontra nenhuma força então

$$\frac{d\mathbf{p}}{dt} = \dot{\mathbf{p}} = 0.$$

Em outros termos, consideremos \mathbf{s} como algum vetor constante, de modo que

$$\dot{\mathbf{p}} \cdot \mathbf{s} = \mathbf{F} \cdot \mathbf{s} = 0.$$

Assim, integrando em relação ao tempo

$$\mathbf{p} \cdot \mathbf{s} = \text{constante},$$

isso expressa o fato de que a *componente da quantidade de movimento linear em uma direção na qual a força se anula é constante no tempo*.

Teorema 1.2.2. *O momento angular de uma partícula não sujeita a qualquer torque é conservado.*

A quantidade de movimento angular de uma partícula em relação a origem é definida como

$$\mathbf{L} = \mathbf{r} \times \mathbf{p}.$$

O torque é definido em relação a mesma origem como

$$\boldsymbol{\tau} = \mathbf{r} \times \mathbf{F},$$

onde \mathbf{r} é o vetor de posição a partir da origem até o ponto onde a força \mathbf{F} é aplicada. Segue que

$$\boldsymbol{\tau} = \mathbf{r} \times m\dot{\mathbf{v}} = \mathbf{r} \times \dot{\mathbf{p}},$$

assim, derivando a equação 1.2.1 tem-se

$$\dot{\mathbf{L}} = \frac{d}{dt}(\mathbf{r} \times \mathbf{p}) = (\dot{\mathbf{r}} \times \mathbf{p}) + (\mathbf{r} \times \dot{\mathbf{p}}),$$

assim

$$(\dot{\mathbf{r}} \times \mathbf{p}) = \dot{\mathbf{r}} \times m\mathbf{v} = m(\dot{\mathbf{r}} \times \dot{\mathbf{r}}) \equiv 0,$$

concluimos que

$$\dot{\mathbf{L}} = \mathbf{r} \times \dot{\mathbf{p}} = \boldsymbol{\tau}.$$

Se nenhum torque estiver atuando no sistema ($\boldsymbol{\tau} = 0$), então $\dot{\mathbf{L}} = 0$ e \mathbf{L} é um vetor constante no tempo.

Teorema 1.2.3. *A energia total E de uma partícula em um campo de força conservativo é constante no tempo.*

Se uma força \mathbf{F} exerce trabalho sobre uma partícula, produzindo um deslocamento de uma posição 1 para uma posição 2, esse trabalho é definido como

$$W_{12} \equiv \int_1^2 \mathbf{F} \cdot d\mathbf{r}. \quad (1.2.2)$$

Se \mathbf{F} é a força líquida resultante atuando sobre a partícula, então

$$\mathbf{F} \cdot d\mathbf{r} = m \frac{d\mathbf{v}}{dt} \frac{d\mathbf{r}}{dt} dt = m \frac{d\mathbf{v}}{dt} \mathbf{v} dt$$

mas pela regra do produto temos

$$m \frac{d\mathbf{v}}{dt} \mathbf{v} = \frac{m}{2} \frac{d}{dt} (\mathbf{v} \cdot \mathbf{v}), \quad (1.2.3)$$

a qual substituindo em (1.2.1) nos dá

$$\mathbf{F} \cdot d\mathbf{r} = \frac{m}{2} \frac{d}{dt}(\mathbf{v} \cdot \mathbf{v}) = \frac{m}{2} \frac{d}{dt}(v^2)dt = d\left(\frac{1}{2}mv^2\right).$$

Então o resultado de (1.2.2) é exato e o trabalho efetuado pela força total \mathbf{F} atuando sobre a partícula é igual a variação de sua energia cinética $T = (1/2)mv^2$.

$$W_{12} = \left(\frac{1}{2}mv^2\right)\Big|_1^2 = \frac{1}{2}m(v_2^2 - v_1^2) = T_2 - T_1, \quad (1.2.4)$$

onde é importante observar que a força \mathbf{F} que leva a equação 1.2.4 é a força resultante.

Para definir a energia potencial em termos de trabalho é necessário mover a partícula sem alteração da energia cinética

$$\int_1^2 \mathbf{F} \cdot d\mathbf{r} \equiv U_1 - U_2, \quad (1.2.5)$$

assim o trabalho ao mover a partícula é a diferença de energia potencial entre os dois pontos. Reescrevamos a equação 1.2.5 em função do gradiente de uma função escalar U da seguinte forma

$$\int_1^2 \mathbf{F} \cdot d\mathbf{r} = - \int_1^2 (\nabla U) \cdot d\mathbf{r} = - \int_1^2 dU = U_1 - U_2.$$

Então, igualando as equações 1.2.4 e 1.2.5 e definindo energia total como $E = T + U$ chega-se a conclusão que

$$T_2 - T_1 = U_1 - U_2$$

$$E_1 = E_2.$$

1.2.2 Equações de movimento para uma partícula

Conhecendo as posições e velocidades das partículas de um sistema mecânico em um determinado instante, pode-se prever um estado futuro do sistema conhecendo as forças que agem determinando unicamente o movimento do sistema. Essa previsão é chamada de *Princípio Determinístico de Newton*. Ocorre que em um referencial inercial a equação de Newton (1.2) pode ser escrita da seguinte forma

$$\mathbf{F} = m\ddot{\mathbf{r}}, \quad (1.2.6)$$

a qual é uma equação diferencial de segunda ordem que pode ser integrada para encontrar o valor $\mathbf{r}(t)$, que descreve o movimento de uma partícula desde que \mathbf{F} seja conhecida, onde \mathbf{F} pode ser uma função da posição, da velocidade e do tempo $\mathbf{F} = \mathbf{F}(\mathbf{r}, \mathbf{v}, t)$.

Exemplo 1.2.4. *O movimento de um partícula sujeita a uma força elástica.*

A equação do movimento do oscilador harmônico simples pode ser obtida substituindo-se a força da Lei de Hooke

$$F = -K\Delta X,$$

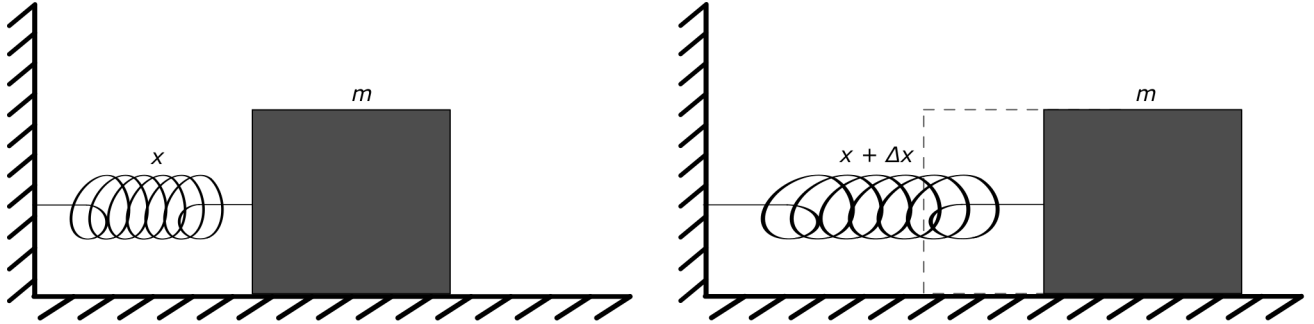


Figura 1.1: Movimento harmônico simples

na equação newtoniana (1.2.6), portanto

$$m\ddot{x} = -K\Delta X, \quad (1.2.7)$$

em que K é uma constante de resistência e ΔX é a variação da partícula de sua posição de repouso.

Fazendo $\omega_0^2 = K/m$ podemos escrever a equação $m\ddot{x} = -Kx$ como

$$\ddot{x} + \omega_0^2 x = 0. \quad (1.2.8)$$

A solução para esta equação diferencial de segunda ordem pode ser expressa de duas formas

$$\begin{aligned} x(t) &= A \sin(\omega_0^2 t - \delta) \\ x(t) &= A \cos(\omega_0^2 t - \phi), \end{aligned}$$

onde δ e ϕ são fases que se diferem por $\pi/2$. Essas soluções revelam a natureza oscilatória do problema.

■

Exemplo 1.2.5. *A equação de movimento de um pêndulo simples.*

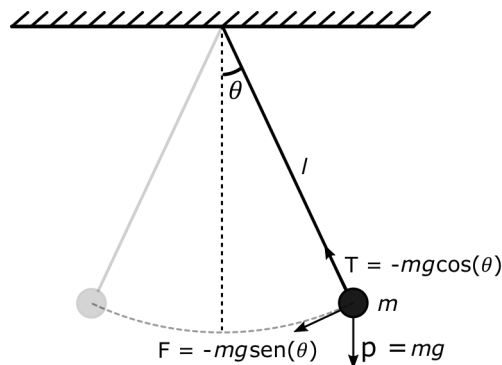


Figura 1.2: Pêndulo planar

Um pêndulo simples, como ilustrado na figura 1.2, é um sistema composto por um corpo que realiza oscilações preso a um extremidade de uma haste ideal. É um sistema não linear com uma força restauradora simétrica proporcional ao seno do ângulo θ e para pequenas oscilações uma aproximação linear é possível de ser utilizada.

A intensidade da força de restauração é dada por $F = -mg \sin\theta$ que quando substituída na equação da Segunda Lei de Newton $\mathbf{F} = m\mathbf{a}$, onde \mathbf{a} é a aceleração angular dada por $\mathbf{a} = \ddot{\theta}l$, onde l é o comprimento da haste do pêndulo, encontra-se a equação de movimento do pêndulo plano

$$\begin{aligned} -mg \sin\theta &= m\ddot{\theta}l \\ \ddot{\theta} + \frac{g}{l} \sin\theta &= 0. \end{aligned} \quad (1.2.9)$$

Podemos inferir a periodicidade desta equação de movimento comparando-a com a equação (1.2.8), fazendo $g/l = \omega_0^2$ e considerando ângulos pequenos para qual podemos aproximar θ (em radianos) pelo valor de seu seno, isto é, $\theta \approx \sin\theta$, como por exemplo, $|\theta| < \pi/18$, onde o erro é menor do que 10^{-3} para esta aproximação.

1.3 Modelagem lagrangiana

A medida que os sistemas ficam mais complexos a modelagem newtoniana vai perdendo sua eficácia pois ela exige a análise de todas as forças atuantes no sistema. Então, a modelagem lagrangiana se mostra mais eficaz porque de forma conveniente faz distinção entre forças ativas, que são consideradas as verdadeiras causas do movimento, e as forças de vínculos, que asseguram a preservação das restrições geométricas e cinemática de um sistema realizando trabalho virtual zero em muitas situações de interesse físico e, assim, permitindo que a técnica utilize menos dados que a modelagem newtoniana. A presente seção mostra como obter as equações de Euler-Lagrange utilizando deslocamentos virtuais e o princípio de D'Alembert, e uma forma de calcular dissipação de energia de um sistema com a função de dissipação de Rayleigh.

1.3.1 Vínculos

Vínculos são restrições de natureza geométrica ou cinemática do movimento das partículas de um sistema, ou seja, eles limitam as posições ou velocidades das partículas e precedem a dinâmica devendo ser levadas em conta para a formulação das equações de movimento.

Exemplo 1.3.1. *Partícula restrita a uma superfície.*

As coordenadas de posição de uma partícula não são independentes e estão relacionadas pela equação da superfície $f(x, y, z) = 0$, ou por $f(x, y, z, t) = 0$ se a superfície está em movimento ou deformando-se.

Se a restrição do movimento das partículas do sistema tiver origem nas equações do movimento, então essa restrição é de natureza dinâmica não sendo um vínculo.

■

Exemplo 1.3.2. *O movimento de uma partícula sob o efeito de uma ação central que ocorre num plano mas não é um vínculo sobre o movimento.*

Este exemplo é diferente de uma partícula se movendo em uma superfície planar. O movimento ocorre num plano devido a atuação das forças, e não por dependências das coordenadas.

■

Se os vínculos forem da forma

$$f(\xi_1, \xi_2, \dots, \xi_n, t) = 0, \quad (1.3.1)$$

onde a função f depende apenas das coordenadas que descrevem as posições do sistema e possivelmente do tempo, são chamados de vínculos holônomos.

Exemplo 1.3.3. *Pêndulo acoplado duplo com hastes rígidas.*

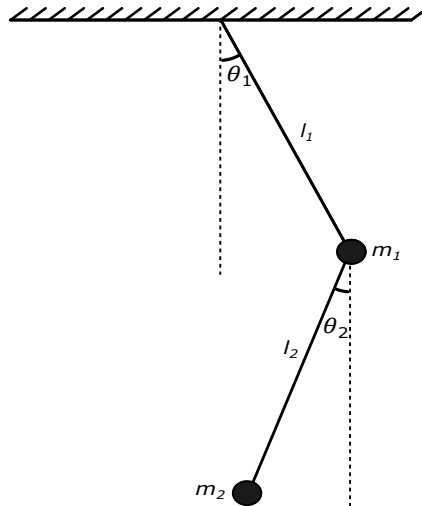


Figura 1.3: Pêndulo duplo

As equações de vínculo do pêndulo duplo são

$$x_1^2 + y_1^2 = l_1^2$$

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 = l_2^2,$$

de acordo com a Figura 1.3, onde (x_1, y_1) e (x_2, y_2) são as coordenadas de m_1 e m_2 , respectivamente, considerando um plano cartesiano com origem na fixação da haste l_1 com a superfície.

■

Em geral, vínculos da forma $f(\xi_1, \xi_2, \dots, \xi_n, \dot{\xi}_1, \dot{\xi}_2, \dots, \dot{\xi}_n, t) = 0$ que dependam da velocidade do sistema não são holônomos, porém podem existir vínculos holônomos que além das posições envolvam a velocidade das partículas, tipicamente em movimento de corpos rígidos sem deslizamento, desde que possam ser reduzidos a forma de vínculos holônomos (1.3.1).

Exemplo 1.3.4. *Vínculo de rolamento de uma moeda rolando sem deslizar sobre um plano.*

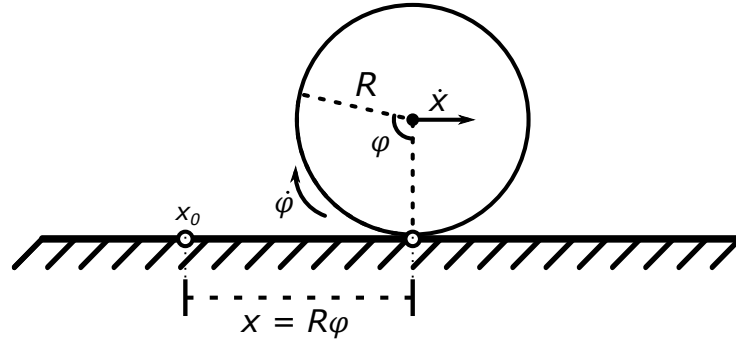


Figura 1.4: Rolamento sem deslizamento

A equação de vínculo desse problema é portanto

$$\dot{x} = R\dot{\varphi},$$

onde \dot{x} é a velocidade de deslocamento do centro de massa e $\dot{\varphi}$ é a velocidade de rotação do cilindro em torno do seu centro de massa. Essa equação diferencial pode ser integrada, logo

$$x = R\varphi,$$

onde x é a posição do centro de massa e φ é o ângulo de rotação em torno do centro de massa.

■

Vínculos não holonomos supõe-se restrições sobre as velocidades possíveis e raramente são integráveis. A importância dos vínculos holônomos é que em problemas onde todos os vínculos são deste tipo existe a possibilidade de introduzir coordenadas generalizadas, esses sistemas são chamados de sistemas holônomos.

1.3.2 Deslocamentos virtuais

O deslocamento virtual $\delta \mathbf{r}$ é o deslocamento que uma partícula do sistema em uma superfície até um ponto infinitamente próximo compatível com os vínculos sobre a superfície no mesmo instante t . O uso da palavra virtual significa que este não é necessariamente o movimento real que acontece com a passagem do tempo. O deslocamento real $d\mathbf{r}$ de uma partícula é o deslocamento que uma partícula do sistema sofre em um intervalo de tempo dt de acordo com as equações de movimento do sistema.

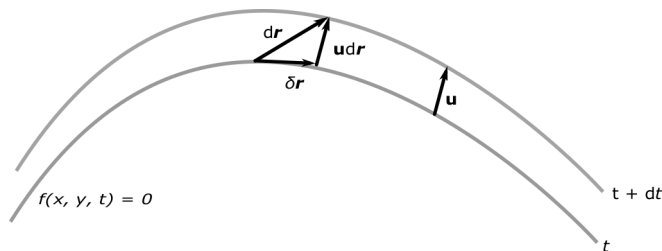


Figura 1.5: Deslocamento virtual

Na Figura 1.5 vemos que o deslocamento virtual ($\delta \mathbf{r}$) é tangente à superfície e que deslocamento real ($d\mathbf{r}$), em geral, não é tangente a superfície. Se a superfície for fixa obtemos $\delta \mathbf{r} = d\mathbf{r}$.

1.3.3 Trabalhos virtuais

Trabalho virtual é chamado o trabalho realizado por um deslocamento virtual. Se a superfície for idealmente lisa a força de vínculo que mantém a partícula sobre a superfície é perpendicular e, portanto, o trabalho realizado por elas em um trabalho virtual é zero.

$$W = \mathbf{F} \cdot \delta \mathbf{r} = 0.$$

As forças de vínculos responsáveis pelo rolamento sem deslizamento de um corpo sobre uma superfície também satisfazem a propriedade do trabalho virtual ser zero. Nessa situação a velocidade do ponto de contato do corpo com a superfície é nula, e como os deslocamentos virtuais são compatíveis com os vínculos, o deslocamento virtual também é nulo e, portanto, não há trabalho virtual. O trabalho no deslocamento real pode não ser zero, a importância de considerar os deslocamentos virtuais vem destas observações.

Caso estático

Em um sistema de partículas em equilíbrio a força resultante em cada partícula i é zero. Por conveniência, distingui-se na força resultante uma parte chamada de Força Aplicada ou Força Ativa e outra parte chamada de Força de Vínculo, isto é,

$$\mathbf{F}_i = \mathbf{F}_i^{(a)} + \mathbf{f}_i, \quad (1.3.2)$$

onde $\mathbf{F}_i^{(a)}$ é a Força Aplicada e \mathbf{f}_i é a Força de Vínculo.

Supondo que os vínculos são ideais, isto é, o trabalho realizado por essas forças quando são realizados deslocamentos virtuais das partículas é zero, obtemos

$$\sum_{i=1}^N \mathbf{f}_i \cdot \delta \mathbf{r}_i = 0. \quad (1.3.3)$$

Para um sistema em equilíbrio sabemos que

$$\sum_i \mathbf{F}_i \cdot \delta \mathbf{r}_i = 0,$$

o qual aplicando na equação (1.3.2) nos dá

$$\sum_i \mathbf{F}_i^{(a)} \cdot \delta \mathbf{r}_i + \sum_i \mathbf{f}_i \cdot \delta \mathbf{r}_i = 0.$$

Por causa de (1.3.3) tem-se que

$$\sum_i \mathbf{F}_i^{(a)} \cdot \delta \mathbf{r}_i = 0,$$

a qual é condição que só envolve forças aplicadas de equilíbrio estático.

Caso dinâmico

Partindo da segunda lei de Newton

$$\dot{\mathbf{p}}_i = \mathbf{F}_i$$

temos pela equação 1.3.2

$$\dot{\mathbf{p}}_i - \mathbf{F}_i^{(a)} = \mathbf{f}_i.$$

Aplicando agora os deslocamentos virtuais temos que

$$\sum_i (\dot{\mathbf{p}}_i - \mathbf{F}_i^{(a)}) \cdot \delta \mathbf{r}_i = \mathbf{f}_i \cdot \delta \mathbf{r}_i$$

e por (1.3.3) concluímos que

$$\sum_i (\dot{\mathbf{p}}_i - \mathbf{F}_i^{(a)}) \cdot \delta \mathbf{r}_i = 0. \quad (1.3.4)$$

Esta última equação é chamada de *Princípio de D'Alembert* e tem vantagens em relação à formulação newtoniana em sistemas mecânicos em movimento por excluir as forças de vínculo.

De acordo com a segunda lei de Newton a força resultante tem que estar sobre cada partícula, ou seja, para escrever as equações de movimento de um sistema de partículas, é necessário escrever o vetor posição de cada partícula, se o sistema estiver sujeito a vínculos serão usadas mais coordenadas que o necessário pois essas coordenadas não são todas independentes.

Na formulação do princípio de D'Alembert ainda falta uma forma de escrever as equações em termos somente de coordenadas independentes que será possível no caso em que todos os vínculos são holônomos.

1.3.4 Coordenadas generalizadas

A formulação newtoniana exige a consideração da força de vínculo nas equações de movimento. A segunda lei de Newton diz que $\mathbf{F} = m\mathbf{a}$ onde \mathbf{F} é a força resultante sobre cada partícula e isso inclui a força de vínculo \mathbf{f} . Se o sistema é composto por N partículas, há uma equação para cada partícula, como cada vetor requer duas ou três componentes, serão necessárias $2N$ ou $3N$ coordenadas para descrever a posição de cada partícula desse sistema. Mas se o sistema está sujeito a vínculos, muitas dessas coordenadas serão redundantes. Nesse sentido coordenadas generalizadas são mais eficientes por não usar as forças de vínculos, elas são um conjunto de quantidades independentes que especificam o estado de um sistema, geralmente são chamadas de q_1, q_2, \dots, q_n , onde n é grau de liberdade do sistema e o número mínimo de coordenadas generalizadas para descrever o sistema, de modo que as propriedades a seguir são satisfeitas:

- i. São independentes entre si;
- ii. Caracterizam unicamente a configuração do sistema;
- iii. Tornam os vínculos identicamente satisfeitos.

Estudemos num exemplo a construção das coordenadas generalizadas.

Exemplo 1.3.5. *Pêndulo duplo*

O sistema tem ao todo quatro coordenadas e duas relações entre elas, logo duas delas são independentes.

As coordenadas generalizadas θ_1 e θ_2 mostradas na Figura 1.3 atendem os requisito i, ii e iii da definição de coordenadas generalizadas. Além disso, elas especificam

a configuração do sistema de pêndulos em um dado momento, ou seja, elas são coordenadas dadas em função do tempo

$$\begin{aligned}\theta_1 &= \theta_1(t) \\ \theta_2 &= \theta_2(t).\end{aligned}$$

A verificação dos itens i e ii são diretas, para o item iii segue que as coordenadas das partículas m_1 e m_2 em função das coordenadas generalizadas são

$$(x_1, y_1) = (l_1 \sin \theta_1, l_1 \cos \theta_1),$$

$$(x_2, y_2) = (l_1 \sin \theta_1 + l_2 \sin \theta_2, l_1 \cos \theta_1 + l_2 \cos \theta_2).$$

Afirmamos que as identidades dos vínculos podem ser verificadas. De fato, $x_1^2 + y_1^2 = (l_1 \cos \theta_1)^2 + (l_1 \sin \theta_1)^2 = l_1^2(\cos^2 \theta_1 + \sin^2 \theta_1) = l_1^2$, a qual nos dá a primeira equação de vínculo. Para a segunda equação, temos que $(x_2 - x_1)^2 + (y_2 - y_1)^2 = (l_1 \sin \theta_1 + l_2 \sin \theta_2 - l_1 \sin \theta_1)^2 + (l_1 \cos \theta_1 + l_2 \cos \theta_2 - l_1 \cos \theta_1)^2 = l_2^2(\cos^2 \theta_2 + \sin^2 \theta_2) = l_2^2$. Assim, a segunda identidade também é verificada, e concluímos que os vínculos são idênticamente satisfeitos.

■

Coordenadas generalizadas podem ser introduzidas em um sistema sempre que todos os vínculos atuantes são holônomos. Suponha um sistema mecânico com N partículas submetido aos p vínculos holônomos

$$\begin{aligned}f_1(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n, t) &= 0 \\ &\vdots \\ f_n(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n, t) &= 0\end{aligned}\tag{1.3.5}$$

onde cada \mathbf{r}_i , $i = 1, \dots, N$, é descrito com 3 coordenadas espaciais, isto é, $\mathbf{r}_i = (x_i, y_i, z_i)$. Desta forma as funções f_i estão definidas de \mathbb{R}^{3N} em \mathbb{R} . Além disso, supõe-se que as f_i são funções de classe C^k , $k \geq 1$. Denote por $U \subset \mathbb{R}^{3N}$ um aberto contendo todas as soluções do sistema acima. Definimos uma função $F : U \subset \mathbb{R}^{3N} \rightarrow \mathbb{R}^p$ por

$$F(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t) = (f_1(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t), \dots, f_p(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t))$$

É claro que F é de classe C^k e, além disso,

$$F(\mathbf{r}_1, \dots, \mathbf{r}_n, t) = (0, \dots, 0).$$

Também temos que

$$F'(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t) = (f'_1(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t), \dots, f'_p(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t)).$$

Agora, para cada f_i , $i = 1, \dots, p$, $f'_i : \mathbb{R}^{3N} \rightarrow \mathbb{R}$ é funcional linear. Logo $f'_i \equiv 0$ ou f'_i é sobrejetivo. Tomamos f'_i sobrejetivo. Isto implica que F' é sobrejetiva. Isto nos leva ao fato que $0 \in \mathbb{R}^p$ é um valor regular da função $F : \mathbb{R}^{3N} \rightarrow \mathbb{R}$. Pelo Teorema 2 do cap. 7 de [6], temos que

$$\begin{aligned}M &= \{x \in \mathbb{R}^{3N} : F(x_1, y_1, z_1, \dots, x_n, y_n, z_n, t) = (0, \dots, 0)\} \\ &= \{(r_1, r_2, \dots, r_n)\}\end{aligned}$$

é uma superfície de classe C^k e dimensão $n = 3N - p$ em \mathbb{R}^{3N} . Desta forma para qualquer ponto em M existe uma parametrização $\phi : V_0 \rightarrow V$ de classe C^k , em que V_0 é um aberto de \mathbb{R}^n e V é um aberto em \mathbb{R}^{3N} . Assim, a parametrização ϕ nos dá o conjunto de variáveis (q_1, q_2, \dots, q_n) tais que $\phi(q_1, q_2, \dots, q_n) = (r_1, r_2, \dots, r_N)$. Isto significa que as variáveis r_i dependem das variáveis q_j , isto é,

$$r_i = r_i(q_1, q_2, \dots, q_n) \quad i = 1, \dots, N. \quad (1.3.6)$$

É comum tomar esta parametrização como algo global, isto é, as variáveis r_i sempre dependem das mesmas variáveis q_j . Em linguagem geométrica as equações (1.3.5) definem uma hiper superfície de dimensão $(n = 3N - p)$ num espaço de dimensão $3N$.

A escolha de um conjunto de coordenadas generalizadas para descrever um sistema não é única e pode-se escolher quaisquer parâmetros independentes que especifiquem completamente o estado do sistema. Essas quantidades não precisam ter dimensões de comprimento e podem, por exemplo, ter dimensões de energia ou mesmo serem adimensionais, a depender do problema.

1.3.5 Equação de Euler-Lagrange

Com a possibilidade introduzir coordenadas generalizadas de forma que a posição das partículas fiquem completamente determinadas quando os vínculos são holônomos, pode-se no princípio de D'Alembert expressar os deslocamentos virtuais $\delta \mathbf{r}_i$, que não são independentes entre si, em termos dos deslocamentos das coordenadas generalizadas δq_i , que são todas independentes entre si.

Partindo da equação (1.3.4) do princípio de D'Alembert

$$\sum_{i=1}^N (m_i \dot{\mathbf{v}}_i - \mathbf{F}_i^{(a)}) \cdot \delta \mathbf{r}_i = 0,$$

e tomando os deslocamentos virtuais em termos de coordenadas generalizadas

$$\delta \mathbf{r}_i = \sum_k \frac{\partial \mathbf{r}_i}{\partial q_k} \delta q_k$$

obtemos

$$\sum_i (m_i \dot{\mathbf{v}}_i - \mathbf{F}_i) \cdot \sum_k \frac{\partial \mathbf{r}_i}{\partial q_k} \delta q_k = 0,$$

do qual deduzimos que

$$\sum_i (m_i \dot{\mathbf{v}}_i) \cdot \sum_k \frac{\partial \mathbf{r}_i}{\partial q_k} \delta q_k - \sum_i \mathbf{F}_i \cdot \sum_k \frac{\partial \mathbf{r}_i}{\partial q_k} \delta q_k = 0. \quad (1.3.7)$$

Nesse ponto, define-se Q_k como a k -ésima componente da força generalizada dada por

$$Q_k \equiv \sum_i \mathbf{F}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k}. \quad (1.3.8)$$

Assim por (1.3.7) e (1.3.8) chega-se a

$$\sum_i \sum_k m_i \dot{\mathbf{v}}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k} \delta q_k = \sum_k Q_k \delta q_k. \quad (1.3.9)$$

Agora, primeiro note que

$$m_i \dot{\mathbf{v}}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k} = \frac{d}{dt} \left(m_i \mathbf{v}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k} \right) - m_i \mathbf{v}_i \cdot \frac{d}{dt} \left(\frac{\partial \mathbf{r}_i}{\partial q_k} \right) \quad (1.3.10)$$

e, também, que derivando (1.3.6) em relação ao tempo tem-se que

$$\frac{d\mathbf{r}_i}{dt} = \sum_k \frac{\partial \mathbf{r}_i}{\partial q_k} \dot{q}_k + \frac{\partial \mathbf{r}_i}{\partial t} = \mathbf{v}_i. \quad (1.3.11)$$

Portanto

$$\frac{\partial \mathbf{v}_i}{\partial \dot{q}_k} = \frac{\partial \mathbf{r}_i}{\partial q_k} \quad (1.3.12)$$

e segue que

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial \mathbf{r}_i}{\partial q_k} \right) &= \sum_l \frac{\partial}{\partial q_l} \left(\frac{\partial \mathbf{r}_i}{\partial q_k} \right) \dot{q}_l + \frac{\partial}{\partial t} \left(\frac{\partial \mathbf{r}_i}{\partial q_k} \right) \\ &= \sum_l \frac{\partial}{\partial q_k} \left(\frac{\partial \mathbf{r}_i}{\partial q_l} \right) \dot{q}_l + \frac{\partial}{\partial q_k} \left(\frac{\partial \mathbf{r}_i}{\partial t} \right) \\ &= \frac{\partial}{\partial q_k} \sum_l \left(\frac{\partial \mathbf{r}_i}{\partial q_l} \dot{q}_l + \frac{\partial \mathbf{r}_i}{\partial t} \right). \end{aligned}$$

Assim por (1.3.11) encontra-se

$$\frac{d}{dt} \left(\frac{\partial \mathbf{r}_i}{\partial q_k} \right) = \frac{\partial \mathbf{v}_i}{\partial q_k}. \quad (1.3.13)$$

Substituindo (1.3.12) e (1.3.13) em (1.3.10) chega-se a

$$m_i \mathbf{v}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k} = \frac{d}{dt} \left(m_i \mathbf{v}_i \cdot \frac{\partial \mathbf{r}_i}{\partial \dot{q}_k} \right) - m_i \mathbf{v}_i \cdot \frac{\partial \mathbf{v}_i}{\partial \dot{q}_k}.$$

Logo, devido a (1.2.3) segue que

$$m_i \mathbf{v}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k} = \frac{d}{dt} \left[\frac{\partial}{\partial \dot{q}_k} \left(\frac{m_i}{2} v_i^2 \right) \right] - \frac{\partial}{\partial q_k} \left(\frac{m_i}{2} v_i^2 \right). \quad (1.3.14)$$

Quando aplicamos (1.3.14) na equação (1.3.9) resulta em

$$\sum_k Q_k \delta q_k = \sum_k \sum_i \left[\frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} \left(\frac{m_i}{2} v_i^2 \right) - \frac{\partial}{\partial q_k} \left(\frac{m_i}{2} v_i^2 \right) \right] \delta q_k.$$

Agora voltando a equação (1.3.7) obtemos as seguintes equações

$$\begin{aligned} \sum_k \sum_i \left[\frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} \left(\frac{m_i}{2} v_i^2 \right) - \frac{\partial}{\partial q_k} \left(\frac{m_i}{2} v_i^2 \right) \right] \delta q_k - \sum_k Q_k \delta q_k &= 0, \\ \sum_k \left[\frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} \left(\sum_i \frac{m_i}{2} v_i^2 \right) - \frac{\partial}{\partial q_k} \left(\sum_i \frac{m_i}{2} v_i^2 \right) - Q_k \right] \delta q_k &= 0. \end{aligned}$$

Seja T a energia cinética do sistema, isto é

$$T = \sum_i \frac{m_i}{2} v_i^2 = T(q, \dot{q}, t), \quad (1.3.15)$$

então

$$\sum_k \left[\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{\partial T}{\partial q_k} - Q_k \right] \delta q_k = 0.$$

Como os δq_k são arbitrários conclui-se que

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{\partial T}{\partial q_k} = Q_k, \quad (1.3.16)$$

com $k = 1, 2, \dots, n$, onde n é o número de grau de liberdade do sistema.

Para forças aplicadas conservativas $\mathbf{F}_i = -\nabla_i V$, onde

$$V = V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n)$$

e, em consequência,

$$-\nabla_i V = - \left(\mathbf{i} \frac{\partial V}{\partial x_i} + \mathbf{j} \frac{\partial V}{\partial y_i} + \mathbf{k} \frac{\partial V}{\partial z_i} \right),$$

segue da equação (1.3.8) que

$$\begin{aligned} Q_k &= \sum_i \mathbf{F}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_k} = \sum_i \left(\mathbf{F}_{ix} \frac{\partial x_i}{\partial q_k} + \mathbf{F}_{iy} \frac{\partial y_i}{\partial q_k} + \mathbf{F}_{iz} \frac{\partial z_i}{\partial q_k} \right) \\ Q_k &= - \sum_i \left(\frac{\partial V}{\partial x_i} \frac{\partial x_i}{\partial q_k} + \frac{\partial V}{\partial y_i} \frac{\partial y_i}{\partial q_k} + \frac{\partial V}{\partial z_i} \frac{\partial z_i}{\partial q_k} \right). \end{aligned}$$

Disso, é claro que

$$Q_k = - \frac{\partial V}{\partial q_k}.$$

Substituindo na equação (1.3.16) chega-se a

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{\partial T}{\partial q_k} = - \frac{\partial V}{\partial q_k},$$

ou, de outra forma,

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{\partial}{\partial q_k} (T - V) = 0.$$

a qual implica em

$$\frac{d}{dt} \left[\frac{\partial}{\partial \dot{q}_k} (T - V) \right] - \frac{\partial}{\partial q_k} (T - V) = 0.$$

Fazendo $L = T - V$, onde L é chamado de função de Lagrange ou lagrangiana, obtemos a equação

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_k} \right) - \frac{\partial L}{\partial q_k} = 0, \quad (1.3.17)$$

a qual é chamada de equação de Lagrange.

A modelagem lagrangiana facilita a formulação das equações de movimento dos sistemas mecânicos, mas essas equações não são simples de resolver porque são equações não lineares e acoplam as coordenadas generalizadas q_1, \dots, q_n e suas derivadas, elas, em geral, não são resolvidas pelos métodos analíticos conhecidos de resolução de equações diferenciais e a aplicação de resoluções numéricas são necessárias.

Exemplo 1.3.6. Pêndulo simples

Com a modelagem lagrangiana θ é usada como coordenada generalizada. De acordo com a figura 1.2 temos que o vetor posição é

$$\mathbf{r} = (l \sin \theta, l \cos \theta), \quad (1.3.18)$$

e o vetor velocidade é dado por

$$\mathbf{v} = \dot{\mathbf{r}} = (l\dot{\theta} \cos \theta, -l\dot{\theta} \sin \theta).$$

Dado que a energia cinética é dada por $T = \frac{1}{2}mv^2$ e que

$$v^2 = \|\mathbf{v}\|^2 = \mathbf{v} \cdot \mathbf{v} = (l\dot{\theta} \cos \theta)^2 + (-l\dot{\theta} \sin \theta)^2 = l^2\dot{\theta}^2,$$

encontramos

$$T = \frac{1}{2}ml^2\dot{\theta}^2.$$

A energia potencial (V) dado por $V = mgh$, em que o valor de h é o valor de y . Logo $V = mgy = -mgl \cos \theta$. Assim a lagrangiana $L = T - V$ fica

$$L = \frac{1}{2}ml^2\dot{\theta}^2 - (-mgl \cos \theta). \quad (1.3.19)$$

Agora, aplicando esta lagrangiana na equação de Euler-Lagrange

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = 0$$

encontramos

$$ml^2\ddot{\theta} + mgl \sin \theta = 0,$$

ou de outra forma

$$\ddot{\theta} + \frac{g}{l} \sin \theta = 0.$$

Esta é a mesma equação (1.2.9) encontrada pela forma newtoniana. ■

1.3.6 Potenciais generalizados

Potenciais generalizados expandem as equações de Lagrange para o caso em que as forças que agem sobre o sistema também dependam da velocidade, desde que essa dependência seja de forma bem especial. Nas equações encontradas com $L = T - V$, para chegar a equação de Lagrange (1.3.17) partiu-se da suposição que as forças aplicadas sobre cada partícula eram provenientes de uma energia potencial e só dependiam das posições das partículas. Porém existem forças importantes da natureza que dependem da velocidade como, por exemplo, a força eletromagnética onde o movimento de uma partícula carregada em um campo magnético tem a força resultante que age sobre ela dependente da velocidade.

Definição 1.3.7. *Um potencial generalizado é uma função $U(q_1, \dots, q_n, t)$ tal que as forças generalizadas podem ser escritas na forma*

$$Q_k = \frac{d}{dt} \left(\frac{\partial U}{\partial \dot{q}_k} \right) - \frac{\partial U}{\partial q_k}. \quad (1.3.20)$$

A partir da equação (1.3.16), se Q_k for uma força generalizada como (1.3.20) então

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{\partial T}{\partial q_k} = \frac{d}{dt} \left(\frac{\partial U}{\partial \dot{q}_k} \right) - \frac{\partial U}{\partial q_k}.$$

Disso

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) - \frac{d}{dt} \left(\frac{\partial U}{\partial \dot{q}_k} \right) - \frac{\partial T}{\partial q_k} + \frac{\partial U}{\partial q_k} = 0.$$

Portanto conseguimos reescrever as equações de Lagrange da seguinte forma

$$\frac{d}{dt} \left[\frac{\partial}{\partial \dot{q}_k} (T - U) \right] - \frac{\partial}{\partial q_k} (T - U) = 0.$$

1.3.7 Função de dissipação de Rayleigh

Casos de dissipação, como por exemplo, o atrito viscoso num líquido ou um objeto se movendo no ar a baixas velocidades, a resistência pode ser considerada aproximadamente proporcional à velocidade das partículas. Forças desse tipo não se encaixam na formulação lagrangiana porque elas teriam que ser conservativas ou admitir um potencial generalizado que dependa das coordenadas e velocidades e possivelmente do tempo. Apesar disso, é possível formular equações de Lagrange incluindo forças dissipativas de forma que as equações fiquem expressas em termos de coordenadas generalizadas arbitrárias.

Se as forças generalizadas são da forma

$$Q_k = \frac{d}{dt} \left(\frac{\partial U}{\partial \dot{q}_k} \right) - \frac{\partial U}{\partial q_k} + Q'_k$$

onde Q'_k representa a parte das forças generalizadas que não provém de potencial generalizado, as equações de movimento tornam-se

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_k} \right) - \frac{\partial L}{\partial q_k} = Q'_k \quad (1.3.21)$$

com $L = T - U$.

Um caso em que Q'_k representa uma força dissipativa, como o atrito viscoso proporcional a velocidade, é dada em componentes cartesianas por: $\mathbf{F}'_{ix} = -k_{ix}V_{ix}$, $\mathbf{F}'_{iy} = -k_{iy}V_{iy}$, $\mathbf{F}'_{iz} = -k_{iz}V_{iz}$, onde \mathbf{F}'_i é a força dissipativa sobre a i -ésima partícula e k_{ix} , k_{iy} , k_{iz} são constantes positivas.

Rayleigh introduziu uma função que facilita o tratamento desta situação definida por

$$\mathcal{F} = \frac{1}{2} \sum_i (k_{ix}v_{ix}^2 + k_{iy}v_{iy}^2 + k_{iz}v_{iz}^2), \quad (1.3.22)$$

de modo que

$$\mathbf{F}'_{ix} = -\frac{\partial \mathcal{F}}{\partial v_{ix}}, \quad \mathbf{F}'_{iy} = -\frac{\partial \mathcal{F}}{\partial v_{iy}}, \quad \mathbf{F}'_{iz} = -\frac{\partial \mathcal{F}}{\partial v_{iz}}.$$

A parte dissipativa das forças generalizadas pode ser então escrita como

$$Q'_k = \sum_i \mathbf{F}'_i \frac{\partial r_i}{\partial q_k} = \sum_i \mathbf{F}'_i \frac{\partial v_i}{\partial \dot{q}_k} = - \sum_i \left(\frac{\partial \mathcal{F}}{\partial v_{ix}} \frac{\partial v_{ix}}{\partial \dot{q}_k} + \frac{\partial \mathcal{F}}{\partial v_{iy}} \frac{\partial v_{iy}}{\partial \dot{q}_k} + \frac{\partial \mathcal{F}}{\partial v_{iz}} \frac{\partial v_{iz}}{\partial \dot{q}_k} \right) = - \frac{\partial \mathcal{F}}{\partial \dot{q}_k}$$

onde foi utilizado (1.3.12). Desse resultado as equações de movimento (1.3.21) resulta em

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_k} \right) - \frac{\partial L}{\partial q_k} + \frac{\partial \mathcal{F}}{\partial \dot{q}_k} = 0. \quad (1.3.23)$$

A função de dissipação de Rayleigh permite escrever as equações de movimento do sistema da mesma forma que (1.3.23) em qualquer sistema de coordenadas generalizadas. Estudemos um exemplo de aplicação da função de dissipação de Rayleigh.

Exemplo 1.3.8. *Pendulo planar com dissipação*

Consideremos que no problema do pêndulo planar do Exemplo 1.3.6 exista uma dissipação proporcional a velocidade do ar, aproximada pela função de Rayleigh.

A equação de Euler-Lagrange desse problema tem a forma

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} + \frac{\partial \mathcal{F}}{\partial \dot{\theta}} = 0.$$

O lagrangiano já foi obtido em (1.3.19). A função de dissipação \mathcal{F} , assumindo que $k_x = k_y = k$ é

$$\mathcal{F} = \frac{1}{2}(kv_x^2 + kv_y^2) = \frac{1}{2}kl^2\dot{\theta}^2$$

Portanto, a equação de movimento do pendulo planar com dissipação de Rayleigh é

$$\ddot{\theta} + \frac{k}{m}\dot{\theta} + \frac{g}{l}\sin\theta = 0. \quad (1.3.24)$$

■

1.4 Cálculo das variações

Apresentamos as deduções das equações de Euler-Lagrange usando o Princípio de d'Alembert, que descreve o movimento por meio de desvios infinitesimais de configurações instantâneas de um sistema. Entretanto, é possível se chegar a equação de Euler,

$$\frac{\partial f}{\partial y} - \frac{d}{dx} \left(\frac{\partial f}{\partial y'} \right) = 0, \quad (1.4.1)$$

através da minimização de funcionais. Um funcional é uma função real que tem como domínio um espaço de funções, ou seja, um funcional associa um número real a cada função do domínio. Em notação matemática:

$$J[y] = \int_{x_1}^{x_2} f(y(x), y'(x), x) dx. \quad (1.4.2)$$

Exemplo 1.4.1. *Um funcional que associa um valor para cada caminho que ligam dois pontos distintos é dado por*

$$s[y] = \int_{x_1}^{x_2} \sqrt{1 + y'(x)^2} dx.$$

o qual é consequência do comprimento do arco infinitesimal no plano ser dado por

$$ds = \sqrt{dx^2 + dy^2} = \sqrt{1 + \left(\frac{dy}{dx} \right)^2} dx$$



O Princípio de Hamilton, ou Princípio da Mínima Ação, enuncia que o movimento real, comparado com outros movimentos possíveis e considerando o movimento do sistema durante um intervalo de tempo finito, é aquele para o qual o valor de uma quantidade, chamada de ação, é definida como

$$S = \int_{t_1}^{t_2} L(q, \dot{q}, t) dt,$$

onde L é a lagrangiana do sistema, é mínima. Fazendo as seguintes mudanças de variáveis $x = t$; $y = q$; $y' = \dot{q}$; $f = L$ e $J = S$ em 1.4.1 e 1.4.2, podemos observar a equivalência com a equação de Euler-Lagrange. O formalismo deste princípio é estudado por uma disciplina da Matemática conhecido como Cálculo das Variações. Omitiremos a exposição, pois, diante do que foi apresentado nesse capítulo, já temos a ferramenta necessária para desenvolver modelos para simulações de oscilações pendulares e elásticas. Em [7], encontramos o desenvolvimento deste formalismo a fim de modelar pêndulos planares acoplados. Encerraremos esta explanação apresentando uma solução para o problema da Braquistócrona.

Exemplo 1.4.2. *Este problema consiste em determinar a curva unindo dois pontos A e B , não pertencentes a uma mesma reta vertical ou horizontal, e que sob a ação da gravidade, uma partícula deslize ao longo dela, sem atrito, no menor tempo possível.*

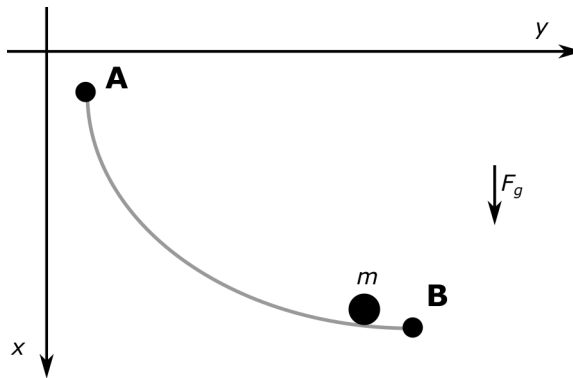


Figura 1.6: Problema da braquistócrona

Definamos $A = (x_1, y_1)$, $B = (x_2, y_2)$ e a origem do sistema sobre ponto A . Além disso, faça o campo de força positivo direcionado no eixo x , vertical, para baixo. Dado que a força atuando na partícula é constante e não existe atrito, afirmamos que o campo é conservativo e que portanto a energia total é $T + V = \text{constante}$. Observe que o potencial no ponto A é $V(x = 0) = 0$ e a partícula inicia o percurso do repouso, logo $T + V = 0$. Disso, vem que $(1/2)mv^2 - mgx = 0$ e assim

$$v = \sqrt{2gx}.$$

O tempo, o qual se quer minimizar, para a partícula se mover de A até B é

$$t = \int_{(x_1, y_1)}^{(x_2, y_2)} \frac{ds}{v} = \int_{x_1}^{x_2} \frac{(dx^2 + dy^2)^{1/2}}{(2gx)^{1/2}} = \int_{x_1=0}^{x_2} \left(\frac{1 + y'^2}{2gx} \right)^{1/2} dx.$$

Podemos retirar a constante $(1/2g)^{1/2}$ da integração, e definir f como

$$f = \left(\frac{1 + y'^2}{x} \right)^{1/2}. \quad (1.4.3)$$

E, por causa de $\partial f / \partial y = 0$, a equação 1.4.1 se torna

$$\frac{d}{dx} \frac{\partial f}{\partial y'} = 0$$

ou

$$\frac{\partial f}{\partial y'} = \text{constante} \equiv (2a)^{-1/2}$$

onde a é uma constante.

Diferenciando (1.4.3) e elevando o resultado ao quadrado, temos

$$\frac{y'^2}{x(1 + y'^2)} = \frac{1}{2a},$$

segue que

$$y' = \left(\frac{x}{2a - x} \right)^{1/2} = \frac{x}{(2ax - x^2)^{1/2}},$$

e, portanto,

$$y = \int \frac{x dx}{(2ax - x^2)^{1/2}}. \quad (1.4.4)$$

Agora, fazendo a alteração de variável,

$$\begin{aligned} x &= a(1 - \cos \theta) \\ dx &= a \sin \theta d\theta, \end{aligned}$$

a integral (1.4.4) torna-se

$$y = \int a(1 - \cos \theta) d\theta = a(\theta - \sin \theta) + \text{constante}.$$

Lembramos agora que as equações paramétricas de uma cicloide passando pela origem são

$$\begin{aligned} x &= a(1 - \cos \theta) \\ y &= a(\theta - \sin \theta). \end{aligned}$$

Justamente a solução encontrada do problema da braquistócrona fazendo a constante de integração igual a zero devido a escolha inicial do ponto A se localizar sobre a origem.

2 Pêndulos planares

Um problema ao estudar as equações de movimento é resolver o sistema de equações diferenciais de segunda ordem gerados, os quais normalmente precisam ser resolvidos por aproximações numéricas. Existe a possibilidade de implementar esses algoritmos por meio de linguagens de programação ou utilizar softwares voltados para computação científica que já trazem esses métodos implementados. Faremos uso do software Mathematica para realizar os cálculos das soluções das equações de movimento encontradas em nosso trabalho. A opção pelo programa justifica-se, principalmente, pela aproximação da escrita matemática com a linguagem utilizada pelo Mathematica, também, por disponibilizar comandos que geram gráficos de funções e animações onde podemos alterar interativamente o valor de variáveis, possibilitando assim, um estudo mais profundo de aplicações de cálculo variacional. Como pré-requisito pedimos conhecimentos básicos sobre programação que são abordados em uma disciplina introdutória na graduação.

Nesse capítulo, usaremos o Mathematica para resolver as equações de movimento dos exemplos do problema do oscilador harmônico simples e do pêndulo planar, ambas desenvolvidas no Capítulo 1, Trataremos ainda um problema de pêndulo planar com haste elástica que resolveremos usando apenas a modelagem lagrangiana.

2.1 Modelagem com o Mathematica

Segundo [8], um algoritmo é um procedimento que descreve uma sequência finita de passos a serem feitos em uma ordem específica cujo objetivo é implementar um procedimento para resolver ou aproximar uma solução de um problema. Este trabalho não descreve os algoritmos numéricos utilizados para solucionar os sistemas de equações diferenciais, utilizamos a função `NDSolve` do Mathematica que conta com várias implementações capazes de identificar e solucionar numericamente equações diferenciais ordinárias e parciais. Mais sobre o comando `NDSolve` pode ser encontrado em [9].

2.1.1 Soluções numéricas

A fim de estudar as soluções das equações de movimento e observar o comportamento dos objetos que estamos interessados seguimos os passos descritos no algoritmo 2.1, que também pode servir de modelo para resolver outros problemas além dos tratados nesse texto.

Algoritmo 2.1 Estudos de sistema de equações de movimento

1) Definir valores de constantes e variáveis

Exemplo: $l=3$; (comprimento da haste)
 $m=1$; (massa)
 $g=-9.81$; (gravidade)

2) Resolver numericamente o sistema de equações

```
solução = NDSolve[{
    sistema de equações diferenciais e valores iniciais
}, { lista das coordenadas generalizadas },
{ domínio do parâmetro }
];
```

3) Gerar gráficos das soluções

Exemplo: Plot[solução]
 ParametricPlot[coordenadas generalizadas do problema] ou
 ParametricPlot3D[coordenadas generalizadas do problema]

Para exemplificar o uso desse algoritmo, desenvolvemos o passo a passo para resolver e mostrar os gráficos da equação de movimento do problema do pêndulo planar do Exemplo 1.3.6. O Código 2.1, também disponível no apêndice A.1, corresponde aos passos 1 e 2, onde na primeira linha temos as atribuições às variáveis l , m e g , que são comprimento da haste, massa e gravidade, respectivamente. Em seguida, a equação de movimento é resolvida para os valores iniciais da coordenada generalizada $\theta(0) = 1$ e velocidade angular $\theta'(0) = 1$, as linhas que seguem indicam que a equação dever ser resolvida para encontrar uma função para a coordenada generalizada θ com parâmetro t variando de 0 a 30 e a solução encontrada pelo programa é atribuída a variável s .

Para escrever no Mathematica códigos que possuam caracteres especiais ou símbolos utilizados na escrita formal matemática, para os quais um teclado padrão não possui um tecla correspondente, como por exemplo o θ (theta), o programa conta com alguns assistentes que auxiliam o usuário. Pode-se abrir um painel contendo os símbolos onde o usuário pode escolher com ajuda do *mouse* ou com o teclado digitando (tecla Esc) + theta + (tecla Esc) ou \[Theta] que automaticamente o editor faz a substituição pelo símbolo. Usamos esta última opção para disponibilizar os códigos no Apêndice para que sejam reproduzidos se houver interesse.

Código 2.1: Solução numérica do pêndulo planar

```

l=3; g=-9.81;
s = NDSolve[
{
  g Sin[θ[t]] + l θ''[t] == 0
  , θ'[0] == 1
  , θ[0] == 1
}
, θ
, {t, 0, 30}
]

```

O resultado da execução é mostrada na Figura 2.1, onde que o programa gera uma função escalar para θ no domínio $[0, 30]$.

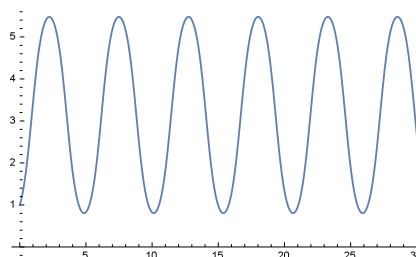
Figura 2.1: Função θ resultado de uma solução numérica do pêndulo planar.

Correspondendo ao passo 3 do algoritmo, que mostra o comportamento da função e do movimento da massa do corpo, usamos os comandos Plot e ParametricPlot para imprimir o gráfico da função e das coordenadas generalizadas. O Código 2.2 desenha o gráfico da função $\theta(t)$ no intervalo $0 \leq t \leq 30$. O comando Evaluate assegura que a função será avaliada para encontrar o valor correspondente ao parâmetro t . O operador ReplaceAll (/.) do Mathematica é utilizado para que a função $\theta[t]$ esteja associada a solução s encontrada pelo NDSolve.

Código 2.2: Plot do gráfico da função θ

```
Plot[Evaluate[θ[t] /. s], {t, 0, 30}]
```

A Figura 2.2 mostra o gráfico da função solução encontrada numericamente pelo programa tem natureza periódica.

Figura 2.2: Gráfico da função θ solução do pêndulo planar

Mas, estamos também interessados em descrever qual caminho a massa do pêndulo percorre. Para isso no Código 2.3, versão em linguagem no Apêndice A.4, usamos

as coordenadas generalizadas que descrevem a posição da massa no plano xy para cada instante t e plotamos em seguida a trajetória. Nesse exemplo, as coordenadas são dadas pela equação 1.3.18 e precisamos da função `ParametricPlot` no Mathematica. O parâmetro t varia de 0 a 30 e acrescentamos o parâmetro `PlotRange` $\rightarrow 3.5$ que indica que ambos os eixos x e y variam de $-3,5$ a $3,5$.

Código 2.3: Trajetória do pêndulo planar

```
ParametricPlot[
  Evaluate[1 Sin[ $\theta[t]$ ], 1 Cos[ $\theta[t]$ ] /. s]
  , {t, 0, 30}
  , PlotRange  $\rightarrow$  3.5
]
```

A Figura 2.3 mostra o caminho percorrido pela massa do pêndulo no intervalo de tempo representado pelo parâmetro t .

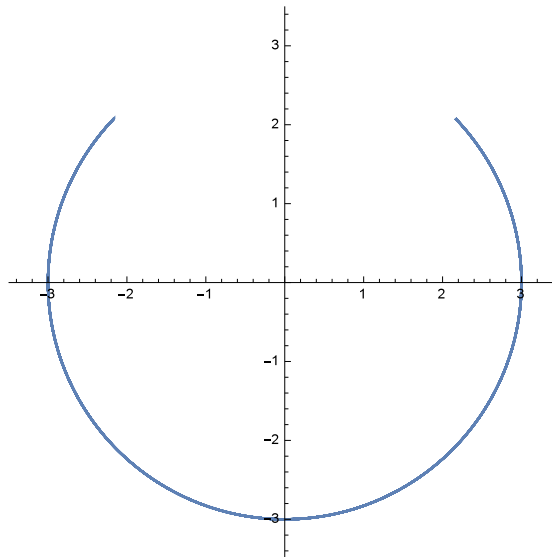


Figura 2.3: Trajetória do pêndulo planar

Fica visível que a massa oscila sobre um caminho e que uma posição pode ser representada por mais de um valor de t no intervalo de tempo. Observar a trajetória dessa forma melhora a compreensão do problema mas ainda pode deixar dúvidas sobre o comportamento da oscilação. Por exemplo, na oscilação com dissipação de energia do exemplo 1.3.8, mantendo as definições de posição e velocidades iniciais e de comprimento de haste como $\theta(0) = 1(\text{rad})$, $\theta'(0) = 1$ e $l = 3$, o ponto inicial do gráfico, à direita, não seria alterado, mas com a perda de energia do sistema a massa não chegaria a mesma altura na esquerda nem ao retornar para a direita completando um ciclo. A oscilação seguiria dissipando energia até a massa parar ao centro no ponto mais baixo da trajetória, e esse último gráfico não traria nenhuma informação sobre esse fenômeno. A Figura 2.4 mostra o comportamento de uma função θ quando um valor de 0,5 é colocado como coeficiente de dissipação na equação 1.3.24.

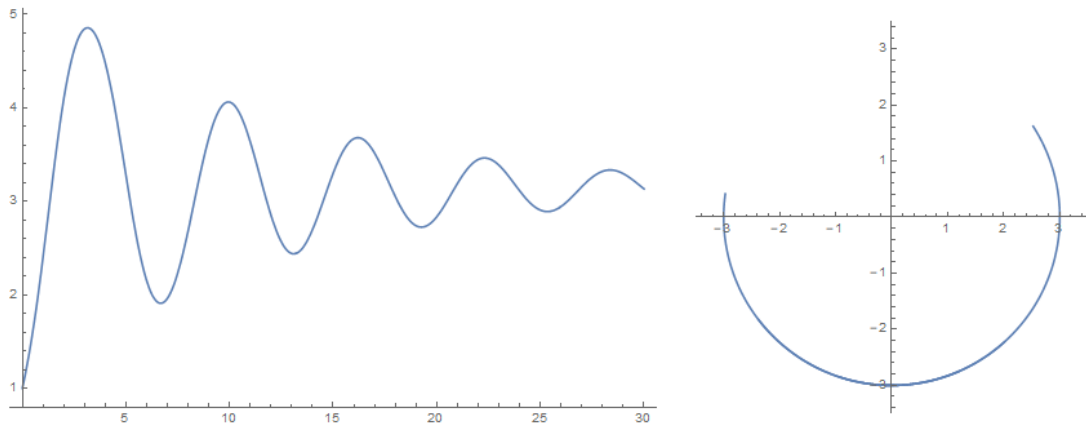


Figura 2.4: Gráficos da função θ e trajetória do pêndulo planar com dissipação

Duas alterações no Código 2.1 foram feitas em relação ao pêndulo planar sem dissipação para produzir esses gráficos, a primeira foi acrescentarmos uma atribuição à constante de dissipação na primeira linha e a segunda foi alterar a equação de movimento resolvida pelo NDSolve levando em conta a função de dissipação de Rayleigh, essas modificações estão no Apêndice A.2.

2.1.2 Simulações

Usamos animações computacionais no Mathematica com a função Manipulate para simular o movimento descrito por um sistema de equações de duas formas, uma por meio de uma curva que cresce e acompanha o caminho percorrido e outra por um disco, no local calculado da massa, ligado à origem por uma reta que representam a massa de um pêndulo e sua haste.

O Código 2.4 é utilizado para fazer o traçado do pêndulo planar usando as mesmas soluções calculadas pelo Código 2.1, mas desenhando em subintervalos do domínio de θ definidos como $[i, i + 0.2]$, onde i está definida em um intervalo $[0, 29.8]$ e é usado pela função Manipulate para gerar varias imagens para fazer o efeito de animação. Esse código também está no Apêndice A.5

Código 2.4: Animação do traçado do pêndulo

```
Manipulate[
  ParametricPlot[
    Evaluate[{1 Sin[ $\theta[t]$ ], 1 Cos[ $\theta[t]$ ]} /. s]
    , {t, i, i + 0.2}
    , PlotRange -> 3.5
  ]
  , {i, 0, 29.8}
]
```

A Figura 2.5 mostra dois valores de i para uma caixa do Manipulate onde o valor pode ser alterado deslizando um controle para esquerda ou direita. Com o intervalo relativamente pequeno, é possível acompanhar, por esse meio, o movimento quando há sobreposição no caminho. Outra forma seria alterar o parâmetro do tamanho do subintervalo e observar o caminho percorrido desde uma posição inicial.

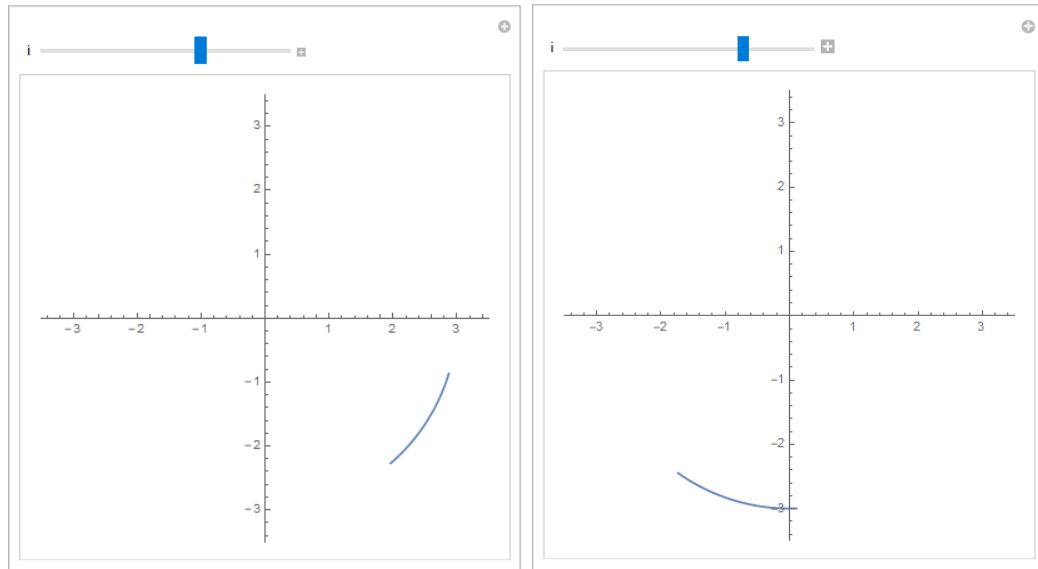


Figura 2.5: Animação do traçado do pêndulo

O Código 2.5 é utilizado para desenhar formas ilustrando um pêndulo nas coordenadas generalizadas do pêndulo planar. Desenhamos um disco de pequeno raio, respeitando a escala utilizada, para representar a massa e uma reta ligando a massa a origem para representar a haste. Para desenhar esses elementos usamos as funções `Disk` e `Line` que são passados para a função `Graphics`. O parâmetro `Axes → True` é necessário porque na função `Graphics` os eixos não são exibidos por padrão.

Código 2.5: Animação do modelo do pêndulo planar

```
Manipulate[
  ParametricPlot[
    {
      Line[{ {0, 0}, Evaluate[{1 Sin[θ[t]], 1 Cos[θ[t]]} /. s][[1]] }],
      Disk[ Evaluate[{1 Sin[θ[t]], 1 Cos[θ[t]]} /. s][[1]], 0.15 ]
    }
    , {t, i, i + 0.2}
    , PlotRange → 3.5
    , Axes → True
  ]
  , {i, 0, 29.8}
]
```

A Figura 2.6 mostra o modelo do pêndulo em duas posições diferentes.

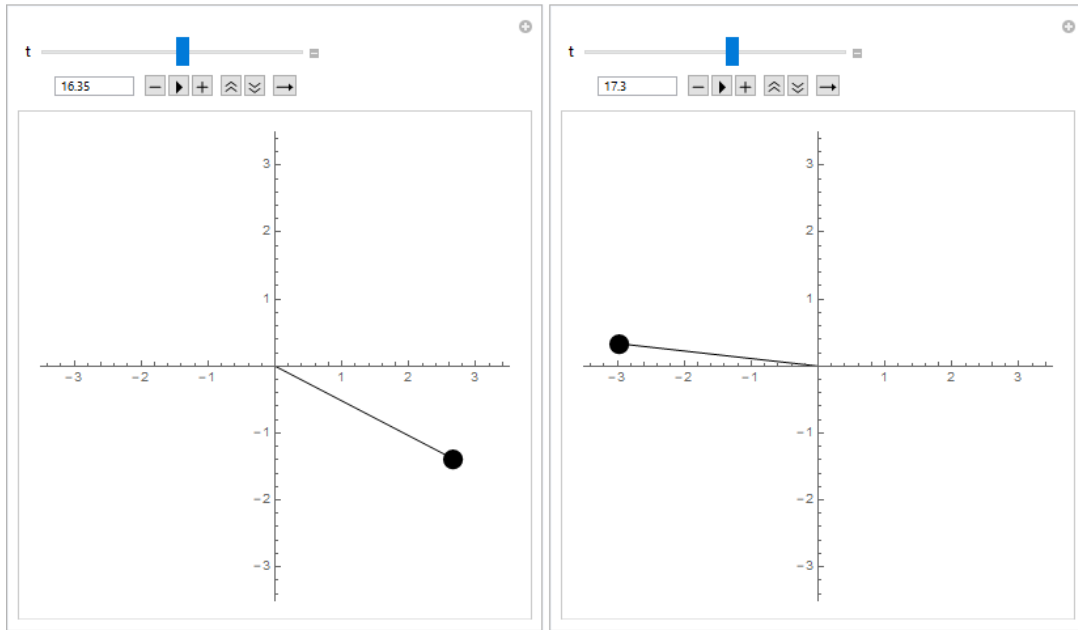


Figura 2.6: Animação do modelo do pêndulo planar

Usamos a função `Show` para combinar as animações dos traçados com as formas que ilustram um pêndulo em uma única imagem. Fazemos isto com objetivo de criar simulações em que podemos manipular, além dos valores do intervalo de tempo t , também os valores das constantes l e m e as condições iniciais do sistema, permitindo assim estudar as consequências provocadas por essas mudanças. Para isso realizamos uma modificação no Algoritmo 2.1. Essas mudanças são mostradas no Algoritmo 2.2. O custo dessas modificações é que a cada nova iteração do `Manipulate` o sistema de equações diferenciais deve ser resolvido. Visando minimizar uma queda no desempenho, o intervalo onde a solução do sistema é procurada deve ser minimizado, calculando somente o necessário para desenhar um quadro.

Algoritmo 2.2 Simulação interativa.

```
g = -9.81;
Manipulate[
  1) resolver numericamente o sistema de equações
  solução = NDSolve[{
    sistema de equações diferenciais
    e valores iniciais como variáveis manipuláveis
  }, {lista das coordenadas generalizadas},
    {domínio do parâmetro}
  ];
  Show [
    2) definir gráficos que compõe as imagens de cada quadro da animação.
    Exemplo: ParametricPlot[coordenadas generalizadas],
             Graphics[Line, Disk] ou Graphics3D[Line, Sphere].
  ]
]
```



```

:
, 3) definir das variáveis manipuláveis e controles do quadro Manipu-
late
:
]

```

Seguindo os passos mostrados desenvolvemos simulações para os pêndulos planares com e sem dissipação, os resultados podem ser vistos na Figura 2.7. Os Códigos encontram-se nos Apêndices A.6 e A.7.

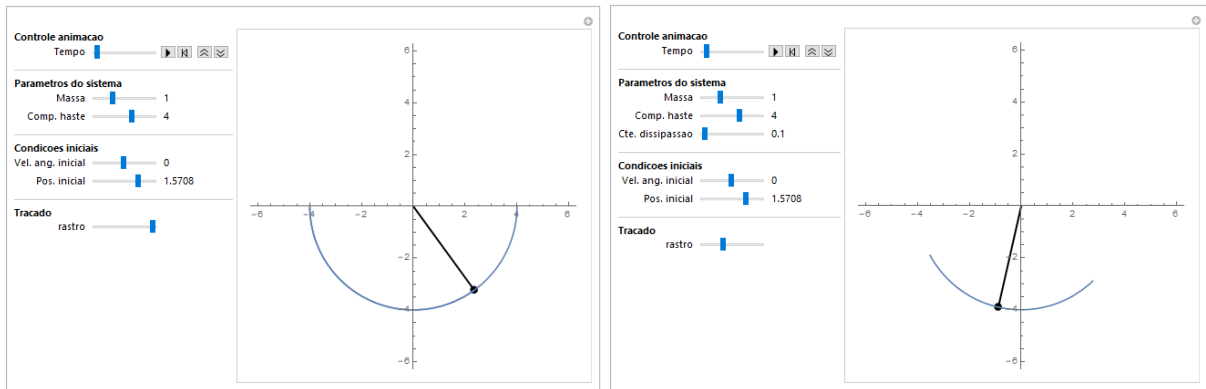


Figura 2.7: Janelas de simulações de pêndulos planares

2.2 Problemas bidimensionais

Nessa seção usaremos o tratamento lagrangiano para modelar dois problemas com elasticidade: o oscilador harmônico simples e o pêndulo planar com haste elástica, veremos também o comportamento desses problemas quando acrescentamos a função de dissipação de Rayleigh.

2.2.1 Modelagem lagrangiana do oscilador harmônico simples

Esse é um tratamento lagrangiano do Exemplo 1.2.4 onde o comprimento de repouso da mola é dado por l e a coordenada generalizada δ representa a variação de comprimento da mola sobre o eixo x . Portanto o vetor posição \mathbf{r} é dado por

$$\mathbf{r} = (l + \delta, 0).$$

Logo o vetor velocidade é

$$\mathbf{v} = \dot{\mathbf{r}} = (\dot{\delta}, 0) \quad (2.2.1)$$

com módulo $v = \dot{\delta}$. Substituindo (2.2.1) na equação de energia cinética (1.3.15) encontra-se que

$$T = \frac{1}{2} m \dot{\delta}^2,$$

e sabendo que a energia potencial elástica é dada por

$$V = \frac{1}{2}K\Delta x^2,$$

onde K é a constante elástica e Δx^2 é a variação do comprimento da haste, conclui-se que

$$V = \frac{1}{2}K\delta^2.$$

Portanto, a lagrangiana é dada por

$$L = T - V = \frac{1}{2}m\dot{\delta}^2 - \frac{1}{2}K\delta^2. \quad (2.2.2)$$

Usamos tal lagrangiana para calcular a equação de Euler-Lagrange com relação a coordenada generalizada δ

$$\frac{\partial L}{\partial \delta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\delta}} \right) = 0.$$

De fato, uma simples conta mostra que

$$m\ddot{\delta} + K\delta = 0$$

ou

$$m\ddot{\delta} = -K\delta$$

a qual é a mesma a equação (1.2.7) calculada pela modelagem newtoniana, com diferença apenas da notação utilizada.

Omitimos o estudo das funções soluções θ desse problema pelo Mathematica, pois, como já observado nos Exemplos 1.2.4 e 1.2.5, elas são semelhantes ao caso do pêndulo planar. A Figura 2.8 mostra a simulação feita no Mathematica. A linha vermelha acima da massa indica o intervalo de deslocamento do objeto. O código está disponível no Apêndice A.8.

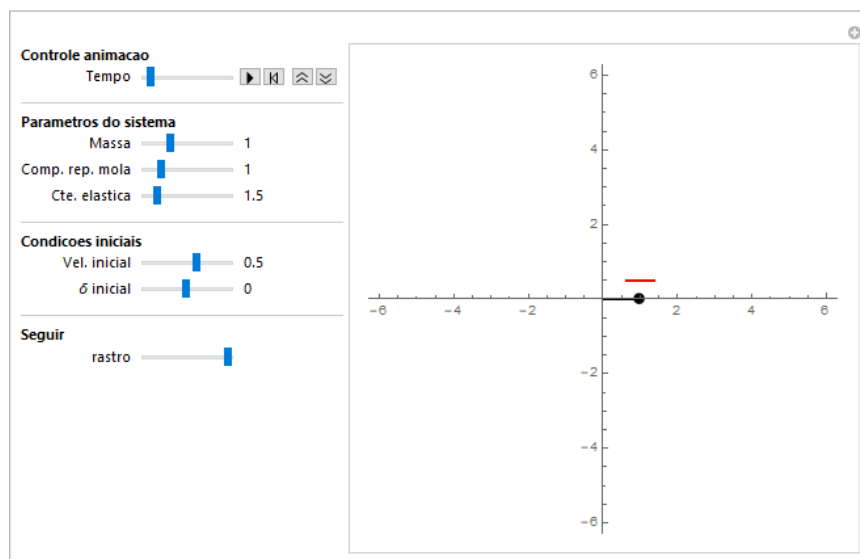


Figura 2.8: Simulação do oscilador harmônico simples

Oscilador harmônico com dissipação

Ao acrescentar a função de dissipação de Rayleigh em um problema, a lagrangiana L permanece inalterada, logo adotamos a L dada pela equação (2.2.2). Por ser um caso bidimensional, desconsidera-se na função de Rayleigh \mathcal{F} , mostrada na equação (1.3.22), a coordenada z , ficando

$$\mathcal{F} = \frac{1}{2} \sum_i (k_{ix} v_{ix}^2 + k_{iy} v_{iy}^2). \quad (2.2.3)$$

Fazendo as constantes de dissipação $k_{ix} = k_{iy} = k$ e substituindo os valores de v , temos que

$$\mathcal{F} = \frac{1}{2} k \dot{\delta}^2.$$

Consequentemente, a equação de Euler-Lagrange para a coordenada generalizada δ

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\delta}} \right) - \frac{\partial L}{\partial \delta} + \frac{\partial \mathcal{F}}{\partial \dot{\delta}} = 0$$

com acréscimo de forças dissipativas é dada pela equação

$$K\delta + k\dot{\delta} + m\ddot{\delta} = 0.$$

A Figura 2.9 mostra a simulação feita para este problema. O código está no Apêndice A.9.

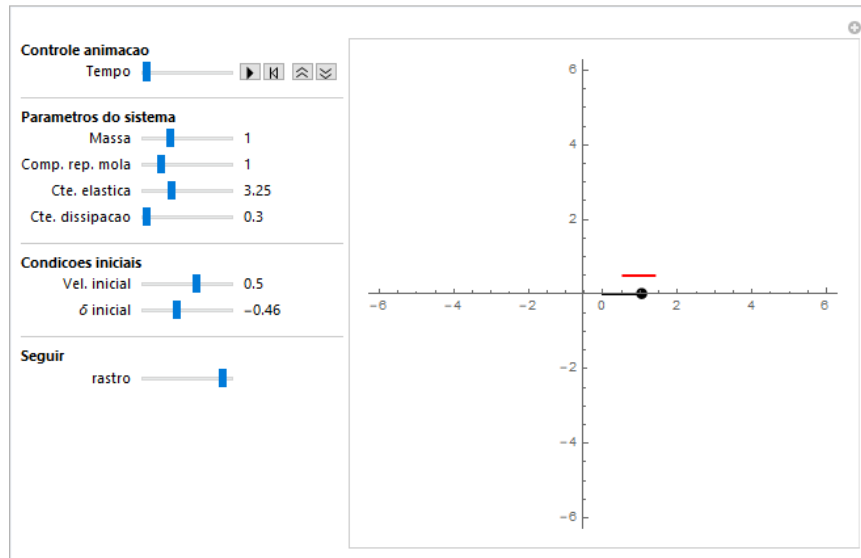


Figura 2.9: Simulação do oscilador harmônico com dissipação

2.2.2 Pêndulo planar com haste elástica

Este exemplo encontra a equação do movimento de um pêndulo onde a haste pode sofrer variação no comprimento segundo uma constante elástica, como ilustrado na Figura 2.10. Segundo [3], este exemplo é mais facilmente obtido pela forma lagrangiana. Todos os problemas que seguem a partir desse ponto serão desenvolvidos apenas pela modelagem lagrangiana.

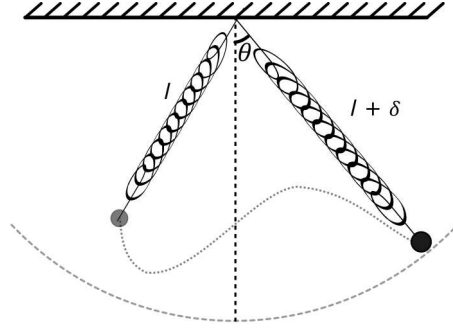


Figura 2.10: Pêndulo planar com haste elástica

As coordenadas generalizadas escolhidas para representar a configuração do sistema são θ para o ângulo formado entre a haste e o eixo vertical e δ que representa a variação de comprimento da haste. O comprimento de repouso da haste elástica é dado por l . Assim, o vetor posição \mathbf{r} é dado por

$$\mathbf{r} = ((l + \delta) \sin \theta, (l + \delta) \cos \theta)$$

e vetor velocidade é dado por

$$\mathbf{v} = \dot{\mathbf{r}} = (\dot{\delta} \sin \theta + \dot{\theta} \cos \theta (l + \delta), \dot{\delta} \cos \theta + \dot{\theta} \sin \theta (l + \delta)). \quad (2.2.4)$$

Disto, a energia cinética é calculada como

$$T = \frac{1}{2} m (\dot{\delta}^2 + (l + \delta)^2 \dot{\theta}^2).$$

Já a energia potencial é obtida pela soma das energias de potencial gravitacional $V_g = mgh$ e potencial elástica $V_e = (1/2)K\Delta x^2$ resultando em

$$V = -mg(l + \delta) \cos \theta + \frac{1}{2} K \delta^2.$$

Obtemos, assim, a lagrangiana

$$L = -\frac{1}{2} K \delta^2 + mg \cos \theta (l + \delta) + \frac{1}{2} m (\dot{\delta}^2 + (l + \delta)^2 \dot{\theta}^2). \quad (2.2.5)$$

Agora resolvendo as equações de Euler-Lagrange em termos das coordenadas generalizadas θ e δ

$$\begin{cases} \frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) = 0 \\ \frac{\partial L}{\partial \delta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\delta}} \right) = 0 \end{cases}$$

encontramos o seguinte sistema de equações de movimento

$$\begin{cases} m(l + \delta)(g \sin \theta + 2\dot{\delta}\dot{\theta} + (l + \delta)^2 \ddot{\theta}) = 0 \\ -gm \cos \theta + K\delta - m(l + \delta)\dot{\theta}^2 + m\ddot{\delta} = 0. \end{cases}$$

O Código 2.6, que corresponde ao Apêndice A.10, é um exemplo usado para estudar uma trajetória do pêndulo com haste elástica. O seu sistema é composto por duas equações e é resolvido para duas coordenadas generalizadas. O gráfico gerado é mostrado na Figura 2.11.

Código 2.6: Trajetória do pêndulo planar com haste elástica

```

m=0.7; g=-9.81; l=1; Ke=10;
s = NDSolve[{
  m (1 +  $\delta[t]$ ) (g Sin[ $\theta[t]$ ] + 2  $\delta'[t]$   $\theta'[t]$  + (1 +  $\delta[t]$ )  $\theta''[t]$ ) == 0,
  -g m Cos[ $\theta[t]$ ] + Ke  $\delta[t]$  - m (1 +  $\delta[t]$ )  $\theta'[t]^2$  + m  $\delta''[t]$  == 0
  ,  $\theta[0]$  == 1.57
  ,  $\theta'[0]$  == 0.5
  ,  $\delta[0]$  == 0.2
  ,  $\delta'[t]$  == 0.1
}
, { $\theta$ ,  $\delta$ }
, {t, 0, 100}
];
inicio = 0;
fim = 5;
PontoInicial = Evaluate[
  {(1 +  $\delta[inicio]$ ) Sin[ $\theta[inicio]$ ]
  , (1 +  $\delta[inicio]$ ) Cos[ $\theta[inicio]$ ]} /. s];
PontoFinal = Evaluate[
  {(1 +  $\delta[fim]$ ) Sin[ $\theta[fim]$ ]
  , (1 +  $\delta[fim]$ ) Cos[ $\theta[fim]$ ]} /. s];
Legendas = PointLegend[{Orange, Purple}
  , {"Ponto Inicial", "Ponto Final"}];
G1 = ParametricPlot[
  Evaluate[{(1 +  $\delta[t]$ ) Sin[ $\theta[t]$ ]
  , (1 +  $\delta[t]$ ) Cos[ $\theta[t]$ ]} /. s]
  , {t, inicio, fim}
  , PlotStyle -> Thick, PlotRange -> Full
  , Axes -> True, AxesOrigin -> {0, 0}
  , PlotLegends -> Legendas
];
G2 = Graphics[{Orange, Disk[PontoInicial[[1]], 0.04]}];
G3 = Graphics[{Purple, Disk[PontoFinal[[1]], 0.04]}];
Show[G1, G2, G3]

```

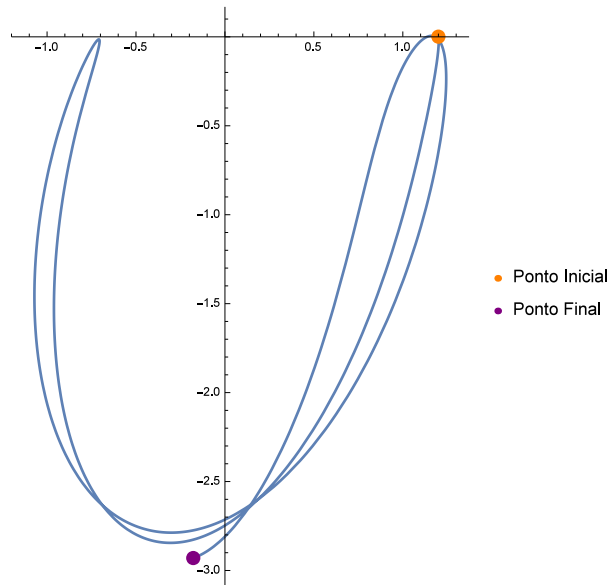


Figura 2.11: Trajetória do pêndulo planar com haste elástica

A Figura 2.12 mostra a simulação feita para esse caso, o código está no Apêndice A.11.

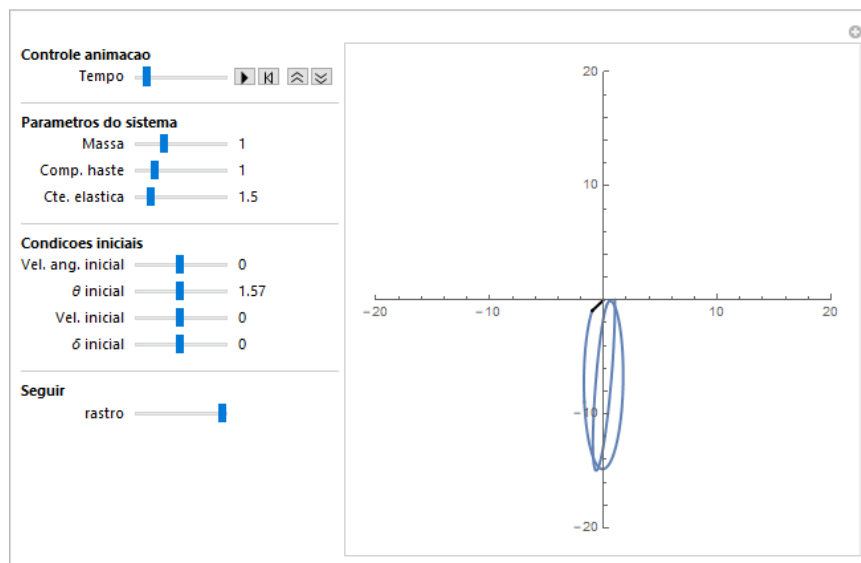


Figura 2.12: Simulação do pêndulo planar com haste elástica

Pêndulo planar com haste elástica e dissipação

A lagrangiana L é dada pela equação (2.2.5). Substituindo a velocidade dada por (2.2.4) na função de dissipação (2.2.3) e considerando que a constante de dissipação é a mesma para as coordenadas x e y , obtemos a equação

$$\mathcal{F} = \frac{1}{2}k(\dot{\delta}^2 + (l + \delta)^2\dot{\theta}^2).$$

Agora para o sistema de equações de Euler-Lagrange com forças dissipativas nas coordenadas generalizadas θ e δ

$$\begin{cases} \frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\theta}} = 0 \\ \frac{\partial L}{\partial \delta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\delta}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\delta}} = 0 \end{cases}$$

chega-se ao sistema de equações de movimento

$$\begin{cases} (l + \delta)((kl + k\delta + 2m\dot{\delta})\dot{\theta} + m(g \sin \theta + (l + \delta)\ddot{\theta})) = 0 \\ k\dot{\delta} + \delta(k - m\dot{\theta}^2) + m(-g \cos \theta - l\dot{\theta}^2 + \ddot{\delta}) = 0. \end{cases}$$

A trajetória do pêndulo com haste elástica e dissipação, Figura 2.13 é estudada no Código A.12.

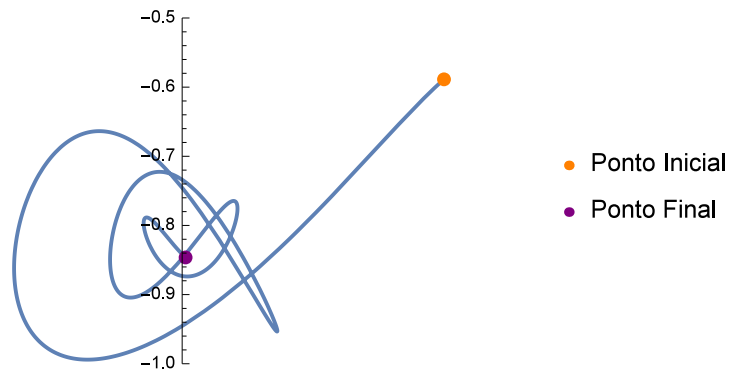


Figura 2.13: Trajetória do pêndulo planar com haste elástica e dissipação

Temos também Código com o comando Manipulate, ver Apêndice A.13, cujo resultado é a Figura 2.14.

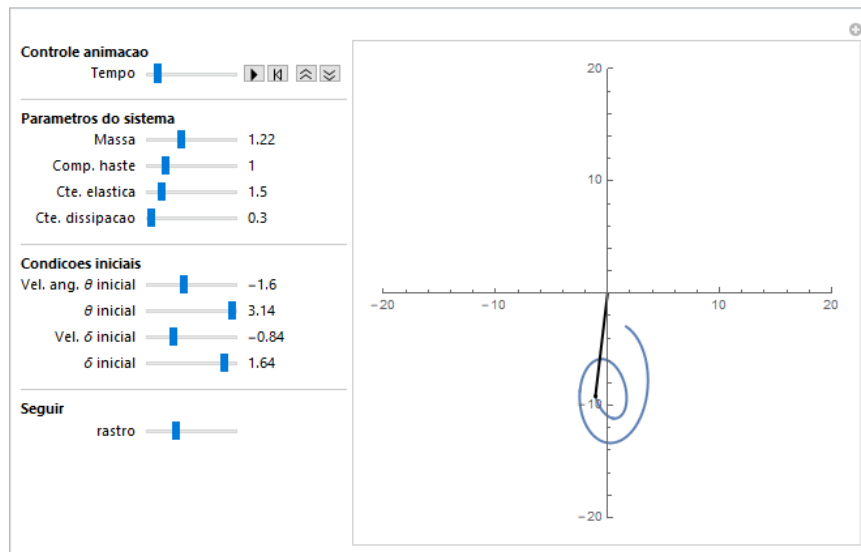


Figura 2.14: Simulação do pêndulo planar com haste elástica e dissipação

3 Pêndulos esféricos

Nesse capítulo desenvolveremos simulações de pêndulos que se movem em espaço com três dimensões. Faremos uso de coordenadas esféricas para descrever a posição de uma massa presa a uma haste. Serão abordados problemas de hastes de tamanho fixo e de tamanho variável segundo uma força elástica e como se comportam esses sistemas quando acrescentamos a função de dissipação de Rayleigh.

As coordenadas generalizadas serão representadas pelas letras gregas θ para o ângulo formado com o plano xy , ϕ para o ângulo formado entre a haste e o eixo z e δ para variação do comprimento da haste. Assim, como nos problemas planares será omitida a dependência do tempo das coordenadas generalizadas. O comprimento da haste elástica será representado por l .

3.1 Pendulo esférico

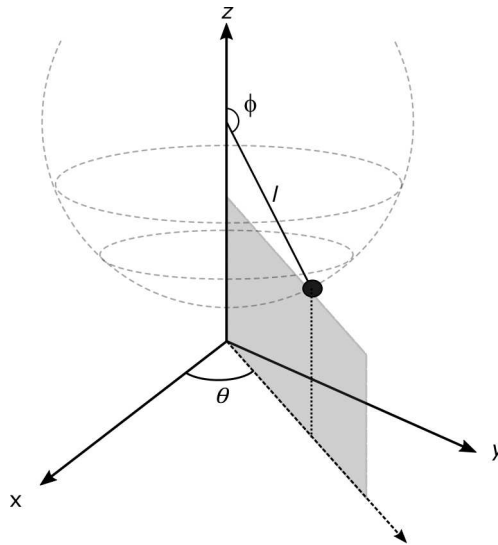


Figura 3.1: Pêndulo esférico

A Figura 3.1 ilustra o movimento de um pêndulo simples com haste de tamanho fixo que se move em um espaço tridimensional. A função de posição \mathbf{r} da massa é descrita por coordenadas esféricas

$$\mathbf{r} = (l \cos \theta \sin \phi, l \sin \theta \sin \phi, l \cos \phi).$$

O vetor velocidade é então

$$\mathbf{v} = \dot{\mathbf{r}} = (-l\dot{\theta} \sin \theta \sin \phi + l\dot{\phi} \cos \theta \cos \phi, l\dot{\theta} \cos \theta \cos \phi + l\dot{\phi} \sin \theta \sin \phi, -l\dot{\phi} \sin \phi). \quad (3.1.1)$$

Já a energia cinética é dada por

$$T = \frac{1}{2}l^2m(\dot{\theta}^2 \sin^2\phi + \dot{\phi}^2)$$

e a energia potencial gravitacional por

$$V = -glm \cos \phi.$$

Em consequência, a lagrangiana será

$$L = \frac{1}{2}l^2m(\dot{\theta}^2 \sin^2\phi + \dot{\phi}^2) + glm \cos \phi.$$

O sistema de equações de Euler-Lagrange para as coordenadas generalizadas θ e ϕ

$$\begin{cases} \frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) = 0 \\ \frac{\partial L}{\partial \phi} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) = 0 \end{cases}$$

nos mostra que o sistema de equações de movimento para o pêndulo esférico é

$$\begin{cases} 2\dot{\theta}\dot{\phi} \cos \phi \sin \phi + \ddot{\theta} \sin^2 \phi = 0 \\ g \sin \phi - l\dot{\theta}^2 \cos \phi \sin \phi + l\ddot{\phi} = 0. \end{cases}$$

No Código 3.1, Apêndice A.14, vemos que há pouca diferença para com os códigos criados para os problemas bidimensionais do capítulo anterior. Somente há mudança no comando utilizado para desenhar a trajetória, de fato utilizamos o comando `ParametricPlot3D`, o qual recebe como parâmetros pontos com três coordenadas.

Código 3.1: Trajetória do pêndulo esférico

```
g=-9.81; l=2;
s = NDSolve[{
  2 θ'[t] ϕ'[t] Cos[ϕ[t]] Sin[θ[t]] + θ''[t] Sin[ϕ[t]]^2 == 0,
  g Sin[ϕ[t]] - l θ'[t]^2 Cos[ϕ[t]] Sin[ϕ[t]] + l ϕ''[t] == 0
, θ[0] == 3.14
, θ'[0] == 1,
, ϕ[0] == 1.6
, ϕ'[0] == 0.1
}
, {θ, ϕ}
, {t, 0, 100}
];
inicio = 0;
fim = 5.5;
PontoInicial = Evaluate[
  {l Cos[θ[inicio]] Sin[ϕ[inicio]]
, l Sin[θ[inicio]] Sin[ϕ[inicio]]
, l Cos[ϕ[inicio]]} /. s];
PontoFinal = Evaluate[
```

```

{1 Cos[ $\theta$ [fim]] Sin[ $\phi$ [fim]]
, 1 Sin[ $\theta$ [fim]] Sin[ $\phi$ [fim]]
, 1 Cos[ $\phi$ [fim]]} /. s;
Legendas = PointLegend[{Orange, Purple}, {"Ponto Inicial", "Ponto Final"}];
G1 = ParametricPlot3D[
  {1 Cos[ $\theta$ [t]] Sin[ $\phi$ [t]], 1 Sin[ $\theta$ [t]] Sin[ $\phi$ [t]], 1 Cos[ $\phi$ [t]]} /. s
, {t, inicio, fim}
, PlotLegends -> Legendas
];
G2 = Graphics3D[{Orange, Sphere[PontoInicial[[1]], 0.05]}];
G3 = Graphics3D[{Purple, Sphere[PontoFinal[[1]], 0.05]}];
Show[G1, G2, G3]

```

Podemos observar o resultado da solução na Figura 3.2.

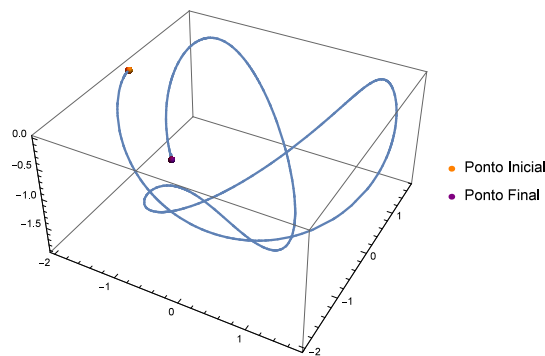


Figura 3.2: Trajetória do pêndulo esférico

Uma simulação foi criada para o deslocamento do pêndulo esférico sobre sua trajetória. O código que descreve tal simulação é encontrado no Apêndice A.15. Na Figura 3.3 encontramos um quadro da simulação em execução. Observamos que foram usados os mesmo elementos do Algoritmo 2.2.

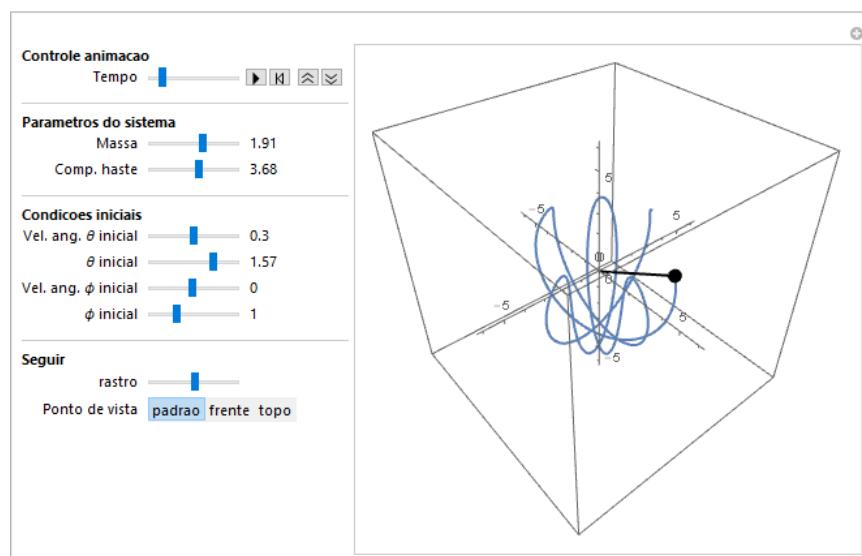


Figura 3.3: Simulação do pêndulo esférico

3.1.1 Pendulo esférico com dissipação

Nesta subseção, vamos introduzir um efeito de dissipação no deslocamento do pêndulo esférico. Usaremos novamente, a função de dissipação de Rayleigh para simular estes fenômenos. A função de dissipação \mathcal{F} dada pela equação (1.3.22) com $k_{ix} = k_{iy} = k_{iz} = k$. Substituindo o vetor velocidade da equação (3.1.1) obtém-se

$$\mathcal{F} = \frac{1}{2}kl^2(\dot{\theta}^2 \sin^2 \phi + \dot{\phi}^2).$$

Assim, o sistema de equações de Euler-Lagrange com forças dissipativas para esse problema é

$$\begin{cases} \frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\theta}} = 0 \\ \frac{\partial L}{\partial \phi} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\phi}} = 0. \end{cases}$$

Deste, chega-se ao sistema de equações diferenciais

$$\begin{cases} l^2 \sin \phi (\dot{\theta} (k \sin \phi + 2\dot{\phi} m \cos \phi) + m\ddot{\theta} \sin \phi) = 0 \\ l(gm \sin \phi - lm\dot{\theta} \cos \phi \sin \phi + kl\dot{\phi} + lm\ddot{\phi}) = 0. \end{cases}$$

A Figura 3.4 é gerada pelo código no Apêndice A.16 que mostra um comportamento do pêndulo com forças dissipativas.

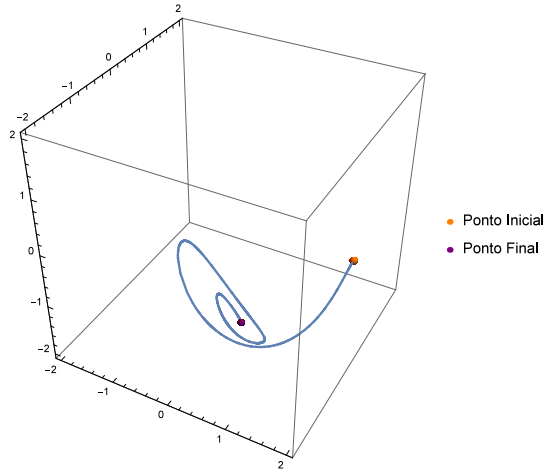


Figura 3.4: Trajetória do pêndulo esférico com dissipação

Uma simulação para esse exemplo, Código A.17, foi criada e é exibida na Figura 3.5.

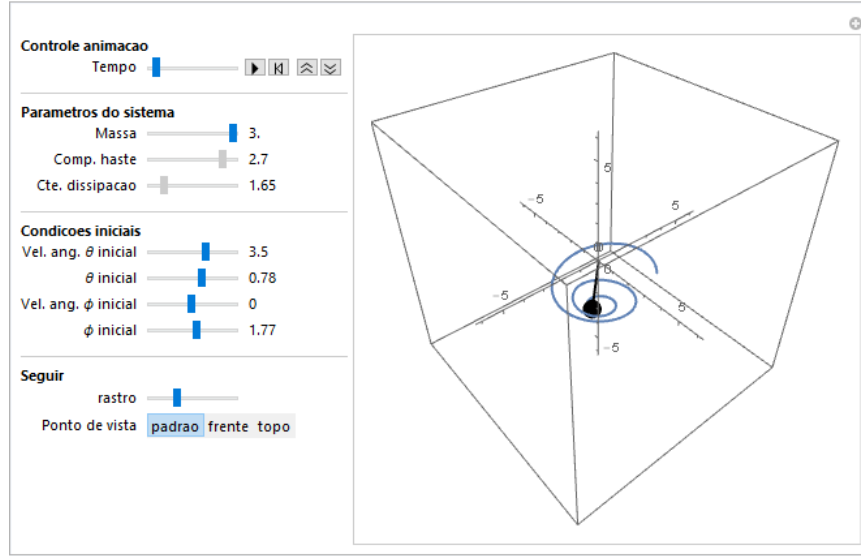


Figura 3.5: Simulação do pêndulo esférico com dissipação

3.1.2 Pêndulo esférico com haste elástica

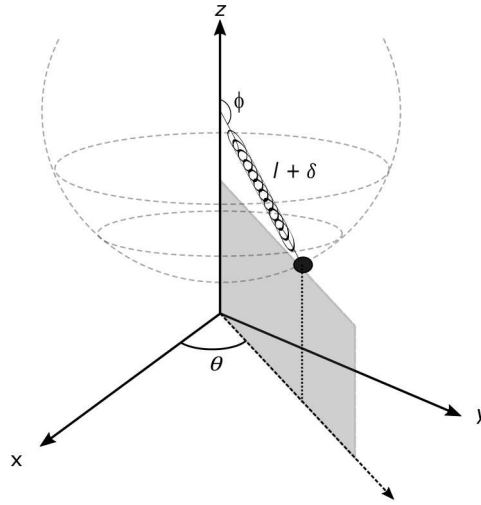


Figura 3.6: Pêndulo esférico com haste elástica

Acrescentando uma variação de comprimento da haste, denominado por δ , segundo uma força elástica ao problema do pêndulo esférico, o vetor de posição \mathbf{r} é dado por

$$\mathbf{r} = ((l + \delta) \cos \theta \sin \phi, (l + \delta) \sin \theta \sin \phi, (l + \delta) \cos \phi).$$

Logo o vetor velocidade será

$$\mathbf{v} = (\dot{\delta} \cos \theta \sin \phi - (l + \delta)(\dot{\theta} \sin \theta \sin \phi - \dot{\phi} \cos \theta \cos \phi), \dot{\delta} \sin \theta \sin \phi + (l + \delta)(\dot{\theta} \cos \theta \sin \phi - \dot{\phi} \cos \phi \cos \theta), \dot{\delta} \cos \phi - (l + \delta)\dot{\phi} \sin \phi).$$

A partir deste encontramos o seguinte valor para a energia cinética,

$$T = \frac{1}{2}m(\dot{\delta}^2 + (l + \delta)^2(\dot{\theta}^2 \sin^2 \phi + \dot{\phi}^2)).$$

Agora adicionando a energia potencial gravitacional ($V_g = mgh$) com a energia potencial elástica ($V_e = (1/2)K\delta^2$) obtemos

$$V = -gm(l + \delta) \cos \phi + \frac{1}{2}K\delta^2.$$

Assim a lagrangiana será descrita por

$$L = \frac{1}{2}m(\dot{\delta}^2 + (l + \delta)^2(\dot{\theta}^2 \sin^2 \phi + \dot{\phi}^2)) + gm(l + \delta) \cos \phi - \frac{1}{2}K\delta^2.$$

Como o nosso problema tem o sistema de Euler-Lagrange em função de θ , ϕ e δ

$$\begin{cases} \frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) = 0 \\ \frac{\partial L}{\partial \phi} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) = 0 \\ \frac{\partial L}{\partial \delta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\delta}} \right) = 0 \end{cases}$$

chegamos no seguinte sistema de equações de movimento

$$\begin{cases} m \sin \phi (l + \delta) (2\dot{\delta}\dot{\theta} \sin \phi + (l + \delta)(2\dot{\theta}\dot{\phi} \cos \phi) + \ddot{\theta} \sin \phi) = 0 \\ m(l + \delta) (-\sin \phi (-g + \cos \phi (l + \delta)\dot{\theta}^2) + 2\dot{\delta}\dot{\phi} + (l + \delta)\ddot{\phi}) = 0 \\ -gm \cos \phi + K\delta - m(l + \delta)(\dot{\theta}^2 \sin^2 \phi + \dot{\phi}^2) + m\ddot{\delta} = 0. \end{cases}$$

Evitando o excesso de repetições, nos limitamos a dizer que criamos o Código A.18, o qual descreve a trajetória do nosso problema. Um exemplo de tal descrição é dada na Figura 3.7 abaixo.

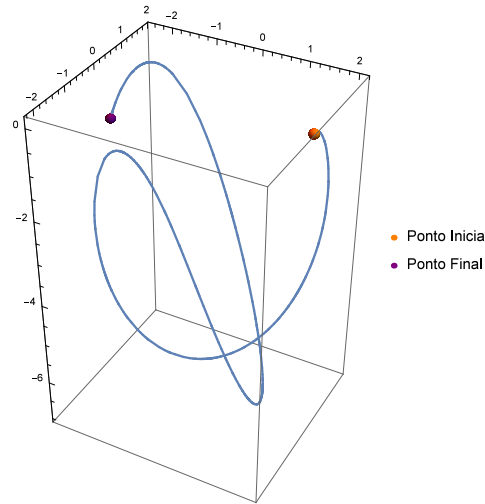


Figura 3.7: Trajetória do pêndulo esférico com haste elástica

Uma janela de uma simulação com o Código A.19 encontrado no apêndice é exibida na Figura 3.8.

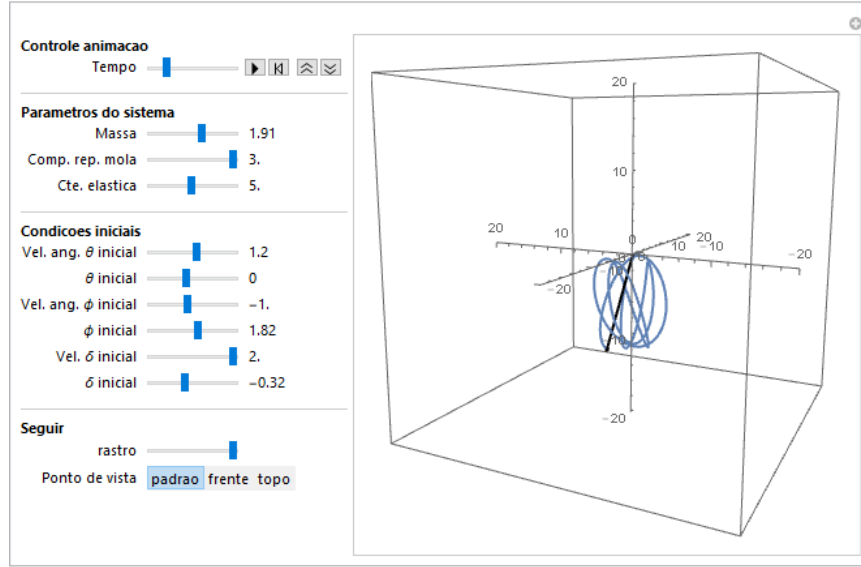


Figura 3.8: Simulação do pêndulo esférico com haste elástica

3.1.3 Pêndulo esférico com haste elástica e dissipação

O nosso último estudo se refere a olhar o pêndulo esférico com haste elástica sobre a influência de um sistema dissipativo. Evitando repetições descrevemos a função de dissipação \mathcal{F} para este problema por

$$F = \frac{1}{2}k(\dot{\delta}^2 + (l + \delta)^2(\dot{\theta}^2 \sin^2 \phi + \dot{\phi}^2)). \quad (3.1.2)$$

Da teoria vemos que as equações de Euler-Lagrange com função dissipativa têm a seguinte forma

$$\begin{cases} \frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\theta}} = 0 \\ \frac{\partial L}{\partial \phi} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\phi}} = 0 \\ \frac{\partial L}{\partial \delta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\delta}} \right) + \frac{\partial \mathcal{F}}{\partial \dot{\delta}} = 0. \end{cases}$$

Aplicando a lagrangiana do problema e a função de dissipação 3.1.2 neste sistema de equações obtemos como o sistema de equações de movimento o seguinte

$$\begin{cases} (l + \delta) \sin \phi (k \dot{\theta} (l + \delta) \sin \phi + 2m \dot{\theta} \dot{\delta} \sin \phi + 2m \dot{\theta} \dot{\phi} (l + \delta) \cos \phi + m(l + \delta) \ddot{\theta} \sin \phi) = 0 \\ (l + \delta) (-m \cos \phi \sin \phi (l + \delta) \dot{\theta}^2 + (kl + k\delta + 2m\delta) \dot{\phi} + m(g \sin \phi + (l + \delta) \ddot{\phi})) = 0 \\ -mg \cos \phi + K\delta + k\dot{\delta} - m(l + \delta)(\dot{\theta}^2 \sin^2 \phi + \dot{\phi}^2) + m\ddot{\delta} = 0. \end{cases}$$

Para a resolução deste sistema construímos o Código A.20, disponível no apêndice. Na Figura 3.9 plotamos o comportamento da resolução do sistema usando o código referido.

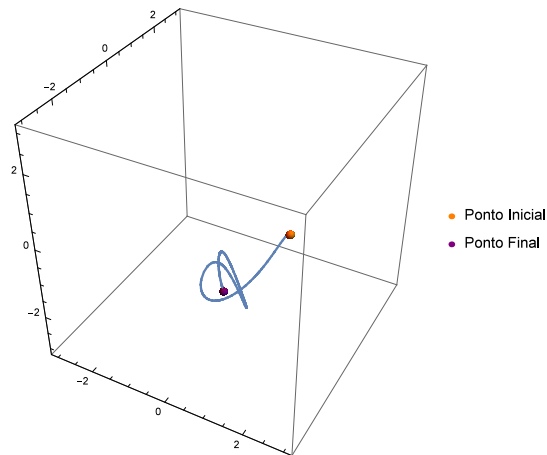


Figura 3.9: Trajetória do pêndulo esférico com haste elástica e dissipação

Finalizamos o nosso trabalho com a criação de um código, intitulado A.21, para simular a trajetória deste pêndulo. Plotamos, na Figura 3.10, uma janela de execução deste código.

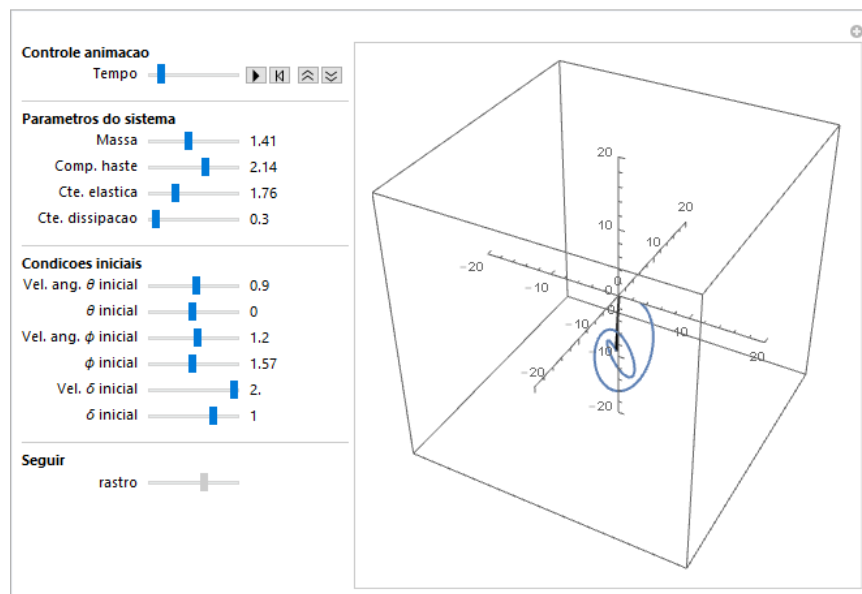


Figura 3.10: Simulação do pêndulo esférico com haste elástica e dissipação

Considerações finais

Da modelagem lagrangiana obtêm-se equações diferenciais de segunda ordem. Mas como saber se o modelo está correto? Conferir a resposta pode funcionar bem em exemplos didáticos, mas deve-se observar que coordenadas diferentes não levam necessariamente ao mesmo conjunto de equações, ou ainda, que o modelo não esteja em nenhum livro. É importante, então, levar a solução além das equações de movimento.

Algoritmos numéricos para solucionar equações diferenciais são difíceis de utilizar sem o uso de computadores, porém, para isso, o conhecimento em programação ainda é necessário e isso costuma ser um problema para estudantes de cursos superiores na área das Ciências Exatas. Esses cursos, contam, geralmente, com pelo menos uma disciplina introdutória de programação, e em um levantamento preliminar feito na Universidade de São Paulo, em [10], mostrou uma taxa de reprovação ou de trancamento de 30%, e em muitos casos superior a 50%, quando o aluno cursa a disciplina pela primeira vez.

Diante disso, as soluções apresentadas neste trabalho visam focar nas soluções dos modelos e menos na programação. Todos os códigos estão disponíveis, onde o leitor pode compreender os modelos sem conhecimento prévio da linguagem do Mathematica, e com poucas alterações outros sistemas mecânicos podem ser simulados, tornando rica a aprendizagem da disciplina.

Referências

- [1] THORNTON, Stephen T.; MARION, Jerry B. **Dinâmica clássica de partículas e sistemas**. Tradução: All Task. 5^a ed. São Paulo: Cengage Learning, 2011.
- [2] LEMOS, Nivaldo A. **Mecânica analítica**. 2^a ed. Rio de Janeiro: Livraria da Física, 2007.
- [3] ROSA, Ricardo M. S. **Modelagens newtoniana, lagrangeana e hamiltoniana de sistemas mecânicos discretos**. Rio de Janeiro: UFRJ, 2008.
- [4] DE AGUIAR, M. A. M. **Tópicos de mecânica clássica**. Rio de Janeiro: Livraria da Física, 2011.
- [5] SUSSMANN, H. J.; WILLEMS, J. C. 300 years of optimal control: from the brachystochrone to the maximum principle. **IEEE Control Systems**, v. 17, p. 32–44, jun. 1997.
- [6] LIMA, Elon Lages. **Análise Real**. 6. ed. Rio de Janeiro: IMPA, 2013. Volume 2.
- [7] MOURA, José Luanderson da Silva. **Um estudo para o sistema mecânico constituído por n pêndulos acoplados e amortecidos pela função de dissipação de Rayleigh**. 2016. Dissertação (mestrado profissional) – UNICAMP, Campinas.
- [8] BURDEN, Richard L.; FAIRES, J. Douglas. **Análise numérica**. Tradução: All Task. 8. ed. São Paulo: Cengage Learning, 2013.
- [9] WOLFRAM RESEARCH, Inc. **Advanced Numerical Differential Equation Solving in the Wolfram Language**. Acesso em: 23 de maio de 2018. Disponível em: <<http://reference.wolfram.com/language/tutorial/NDSolveOverview.html>>.
- [10] BOSSE, Yoram; GEROSA, Marco Aurélio. Reprovações e Trancamentos nas Disciplinas de Introdução à Programação da Universidade de São Paulo : Um Estudo Preliminar. **WEI - Workshop sobre Educação em Computação**, p. 1–10, 2015.

Apêndice A

Projeto Computacional

Com os resultados obtidos dos exemplos dos Capítulos 2 e 3 foram feitas simulações através do programa computacional *Wolfram Mathematica* em sua edição 10.2 *Student Edition* para modelar animações interativas para sistemas mecânico amortecidos ou não pela função de dissipação de Rayleigh, onde o estudante pode manipular os valores de posições, velocidades, massas e constantes para observar os efeitos provocados pelas alterações. As simulações fazem uso, para a resolução destas equações, de comandos numéricos a partir de certas condições iniciais do problema.

Os códigos foram testados em ambientes com sistemas operacionais Windows 10 e Linux Ubuntu 16.04 LTS e podem ser copiados da versão digital PDF deste documento e colados no programa *Wolfram Mathematica*. Eles também estão publicados na Internet pela ferramenta *Google Drive* e podem ser obtidos nos formatos *Notebook* (extensão **.nb**), utilizado com o *Wolfram Mathematica*, ou no Formato de Documento Computável (CDF em inglês, com extensão **.cdf**) utilizado pelo *Wolfram CDF Player*, que permite visualizar os arquivos criados com o *Mathematica*, porém, sem permissão para editá-los. Optou-se em não utilizar acentuação da língua portuguesa nos códigos para evitar problemas de compatibilidade ao fazer a cópia nos diferentes sistemas operacionais.

Código A.1: NDSolve - Pêndulo planar

```
l=3; g=-9.81;
s = NDSolve[
{
  g Sin[\[Theta][t]] + l \[Theta]''[t] == 0
  ,\[Theta]'[0] == 1
  ,\[Theta][0] == 1
}
, \[Theta]
, {t, 0, 30}
]
```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.2: NDSolve: Pêndulo planar com dissipação

```
l=3; m=1; Kd=0.5; g=-9.81;
s = NDSolve[
```

```
{
  (g / l) Sin[\[Theta][t]] + (Kd / m) \[Theta]'[t]
    + l \[Theta]''[t] == 0,
  \[Theta]'[0] == 1,
  \[Theta][0] == 1
}
, \[Theta]
, {t, 0, 30}
]
```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.3: Plot: Solução das equações de movimento planares com ou sem dissipação

```
Plot[Evaluate[\[Theta][t] /. s], {t, 0, 30}]
```

Código A.4: ParametricPlot: Trajetória dos pêndulos planares com ou sem dissipação

```
ParametricPlot[
  Evaluate[{1 Sin[\[Theta][t]], 1 Cos[\[Theta][t]]} /. s]
, {t, 0, 30}
, PlotRange -> 3.5
]
```

Código A.5: Comando Manipulate

```
Manipulate[
  ParametricPlot[
    Evaluate[{1 Sin[\[Theta][t]], 1 Cos[\[Theta][t]]} /. s]
, {t, i, i + 0.2}
, PlotRange -> 3.5
]
, {i, 0, 29.8}
]
```

Código A.6: Pêndulo plano

```
Clear[g, l, m, t, te];
g = -9.81;
Manipulate[
  s = NDSolve[{
    l m (g Sin[\[Theta][te]] + l \[Theta]''[te]) == 0,
    \[Theta]'[0] == d\[Theta]0,
    \[Theta][0] == \[Theta]0
  }
, \[Theta]
, {te, Max[-1, t - 9], t + 1}
```

```

];
Show[
  Graphics[{Thick, Black,
    Line[{0, 0},
      Evaluate[{1 Sin\[Theta][t], 1 Cos\[Theta][t]} /. s][[1]]},
    Disk[Evaluate[{1 Sin\[Theta][t], 1 Cos\[Theta][t]} /. s][[1]],
      m 0.15]
  ],
  PlotRange -> 6.3,
  Axes -> True, AxesOrigin -> {0, 0},
  ImageSize -> Large
],
ParametricPlot[
  Evaluate[{1 Sin\[Theta][i], 1 Cos\[Theta][i]} /. s],
  {i, Max[0.01, t - r], t},
  PlotRange -> 5]
],
Style["Controle animacao", Bold],
{{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
AppearanceElements -> {"PlayPauseButton", "ResetButton",
"FasterSlowerButtons", "ProgressSlider"}, Trigger,
Appearance -> "Labeled", AnimationRate -> 1},
Delimiter,
Style["Parametros do sistema", Bold],
{{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
Appearance -> "Labeled"},
{{l, 4, "Comp. haste"}, 0.5, 6, Slider, ImageSize -> Tiny,
Appearance -> "Labeled"},
Delimiter,
Style["Condicoes iniciais", Bold],
{{d\[Theta]0, 0, "Vel. ang. inicial"}, -15, 15, Slider,
ImageSize -> Tiny, Appearance -> "Labeled"},
{{\[Theta]0, 1.5708, "Pos. inicial"}, -3.14, 3.14, Slider,
ImageSize -> Tiny, Appearance -> "Labeled"},
Delimiter,
Style["Tracado", Bold],
{{r, 7, "rastro"}, 0.01, 7, Slider, ImageSize -> Tiny},
ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.7: Pêndulo plano com dissipação

```

Clear[g, l, m, Kd, t, te];
g = -9.81;
Manipulate[
  s = NDSolve[{
    (g/l) Sin\[Theta][te] + (Kd/m) \[Theta]'[te] + \[Theta]''[te] ==

```

```

0
, \[Theta]'[0] == d\[Theta]0
, \[Theta][0] == \[Theta]0
}
, \[Theta]
, {te, Max[-1, t - 9], t + 1}
];
Show[
Graphics[{Thick, Black,
Line[{0, 0},
Evaluate[{1 Sin\[Theta][t], 1 Cos\[Theta][t]} /. s][[1]]
}],
Disk[
Evaluate[{1 Sin\[Theta][t], 1 Cos\[Theta][t]} /. s][[1]],
m 0.15]
]
, PlotRange -> 6.3
, Axes -> True, AxesOrigin -> {0, 0}
, ImageSize -> Large
]
, ParametricPlot[
Evaluate[{1 Sin\[Theta][i], 1 Cos\[Theta][i]} /. s]
, {i, Max[0.01, t - r], t}
, PlotRange -> 5
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
AppearanceElements -> {"PlayPauseButton", "ResetButton",
"FasterSlowerButtons", "ProgressSlider"}, Trigger,
Appearance -> "Labeled", AnimationRate -> 1}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
Appearance -> "Labeled"}
, {{l, 1, "Comp. haste"}, 0.5, 6, Slider, ImageSize -> Tiny,
Appearance -> "Labeled"}
, {{Kd, 0.1, "Cte. dissipacao"}, 0, 3, Slider,
ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Theta]0, 0, "Vel. ang. inicial"}, -15, 15, Slider,
ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Theta]0, 1.5708, "Pos. inicial"}, -3.14, 3.14, Slider,
ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Tracado", Bold]
, {{r, 2.5, "rastros"}, 0.01, 7, Slider, ImageSize -> Tiny}

```

```
, ControlPlacement -> Left
]
```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.8: Oscilação elástica

```
Clear[g, l, m, Ke, t, te];
g = -9.81;
Manipulate[
  s = NDSolve[{
    Ke \[Delta][te] + m \[Delta]''[te] == 0,
    \[Delta]'[0] == d\[Delta]0,
    \[Delta][0] == \[Delta]0
  }
  , \[Delta]
  , {te, Max[-1, t - 9], t + 1}
];
Show[
  Graphics[{Thick, Black,
    Line[{0, 0}, Evaluate[{(1 + \[Delta][t]), 0} /. s][[1]]},
    Disk[Evaluate[{(1 + \[Delta][t]), 0} /. s][[1]], m 0.15]
  }
  , PlotRange -> 6.3
  , Axes -> {True, True}
  , ImageSize -> Large
]
, ParametricPlot[
  Evaluate[{(1 + \[Delta][i]), 0.5} /. s]
  , {i, Max[0.01, t - r], t}
  , PlotStyle -> {Thick, Red}
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
  AppearanceElements -> {"PlayPauseButton", "ResetButton",
    "FasterSlowerButtons", "ProgressSlider"}, Trigger,
  Appearance -> "Labeled", AnimationRate -> 1}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. rep. mola"}, 0.5, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Ke, 1.5, "Cte. elastica"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Delta]0, 0.5, "Vel. inicial"}, -2, 2, Slider,
```

```

    ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Delta]0, 0, "\[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 7, "rastro"}, 0.01, 7, Slider, ImageSize -> Tiny}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.9: Oscilação elástica com dissipação

```

Clear[g, l, m, Ke, t, te];
g = -9.81;
Manipulate[
  s = NDSolve[{
    Ke \[Delta][te] + Kd \[Delta]'[te] + m \[Delta]''[te] == 0,
    \[Delta]'[0] == d\[Delta]0,
    \[Delta][0] == \[Delta]0
  }
  , \[Delta]
  , {te, Max[-1, t - 9], t + 1}
];
Show[
  Graphics[{Thick, Black,
    Line[{0, 0}, Evaluate[{(1 + \[Delta][t]), 0} /. s][[1]]},
    Disk[Evaluate[{(1 + \[Delta][t]), 0} /. s][[1]], m 0.15]
  }
  , PlotRange -> 6.3
  , Axes -> {True, True}, AxesOrigin -> {-0.55, 0}
  , ImageSize -> Large
]
, ParametricPlot[
  Evaluate[{(1 + \[Delta][i]), 0.5} /. s]
  , {i, Max[0.01, t - r], t}
  , PlotStyle -> {Thick, Red}
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
  AppearanceElements -> {"PlayPauseButton", "ResetButton",
    "FasterSlowerButtons", "ProgressSlider"}, Trigger,
  Appearance -> "Labeled", AnimationRate -> 1}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. rep. mola"}, 0.5, 3, Slider, ImageSize -> Tiny,

```

```

    Appearance -> "Labeled"}
, {{Ke, 1.5, "Cte. elastica"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Kd, 0.3, "Cte. dissipacao"}, 0, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Delta]0, 0.5, "Vel. inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Delta]0, 0, "\[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 2.5, "rastro"}, 0.01, 7, Slider, ImageSize -> Tiny}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.10: Trajetória pêndulo com haste elástica

```

Clear[m, g, l, Ke, s, t]
m = 0.7; g = -9.81; l = 1; Ke = 10;
s = NDSolve[{
  m (1 + \[Delta][t]) (g Sin\[Theta][t] + 2 \[Delta]'[t] \[Theta]'[t] +
    (1 + \[Delta][t]) \[Theta]''[t]) == 0,
  -g m Cos\[Theta][t] + Ke \[Delta][t] - m (1 + \[Delta][t])
    \[Theta]'[t]^2 + m \[Delta]''[t] == 0,
  \[Theta][0] == 1.57,
  \[Theta]'[0] == 0.5,
  \[Delta][0] == 0.2,
  \[Delta]'[0] == 0.1
},
, {\[Theta], \[Delta]}
, {t, 0, 100}
];
inicio = 0;
fim = 5;
PontoInicial = Evaluate[{(1 + \[Delta][inicio]) Sin\[Theta][inicio]
, (1 + \[Delta][inicio]) Cos\[Theta][inicio]} /. s];
PontoFinal = Evaluate[{(1 + \[Delta][fim]) Sin\[Theta][fim]
, (1 + \[Delta][fim]) Cos\[Theta][fim]} /. s];
Legendas = PointLegend[{Orange, Purple}
, {"Ponto Inicial", "Ponto Final "});
G1 = ParametricPlot[Evaluate[
  {(1 + \[Delta][t]) Sin\[Theta][t], (1 + \[Delta][t])
    Cos\[Theta][t]} /. s]
, {t, inicio, fim}
, PlotStyle -> Thick, PlotRange -> Full

```



```

, Axes -> {True, True}
, AxesOrigin -> {0, 0}
, PlotLegends -> Legendas
];
G2 = Graphics[{Orange, Disk[PontoInicial[[1]], 0.04]}];
G3 = Graphics[{Purple, Disk[PontoFinal[[1]], 0.04]}];
Show[G1, G2, G3]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.11: Pêndulo plano com haste elástica

```

Clear[l, m, Ke, t, te];
g = -9.81;
Manipulate[
  s = NDSolve[{
    m (1 + \[Delta][te]) (g Sin\[Theta][te] +
      2 \[Delta]'[te] \[Theta]'[te] +
      (1 + \[Delta][te]) \[Theta]''[te]) == 0,
    -g m Cos\[Theta][te] + Ke \[Delta][te] -
      m (1 + \[Delta][te]) \[Theta]'[te]^2 + m \[Delta]''[te] == 0
    , \[Theta]'[0] == d\[Theta]0
    , \[Theta][0] == \[Theta]0
    , \[Delta]'[0] == d\[Delta]0
    , \[Delta][0] == \[Delta]0
  }
    , {\[Theta], \[Delta]}
    , {te, Max[-1, t - 30], t + 1}
  ];
  Show[
    ParametricPlot[
      Evaluate[{(1 + \[Delta][i]) Sin\[Theta][i],
        (1 + \[Delta][i]) Cos\[Theta][i]} /. s]
      , {i, Max[0.01, t - r], t}
      , PlotStyle -> {Thick}, PlotRange -> 20
      , Axes -> {True, True}, AxesOrigin -> {0, 0}
      , ImageSize -> Large
    ]
    , Graphics[{Thick, Black,
      Line[{0, 0},
        Evaluate[{(1 + \[Delta][t]) Sin\[Theta][t],
          (1 + \[Delta][t]) Cos\[Theta][t]} /. s][[1]]},
      Disk[Evaluate[{(1 + \[Delta][t]) Sin\[Theta][t],
        (1 + \[Delta][t]) Cos\[Theta][t]} /. s][[1]],
        m 0.15]
    ]
  ]
  , Style["Controle animacao", Bold]

```

```

, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
  AppearanceElements -> {"PlayPauseButton", "ResetButton",
    "FasterSlowerButtons", "ProgressSlider"}, Trigger,
  Appearance -> "Labeled", AnimationRate -> 2}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. haste"}, 0.5, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Ke, 1.5, "Cte. elastica"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Theta]0, 0, "Vel. ang. inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Theta]0, 1.57, "\[Theta] inicial"}, 0, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d\[Delta]0, 0, "Vel. inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Delta]0, 0, "\[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 30, "rastro"}, 0.01, 30, Slider, ImageSize -> Tiny}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.12: Trajetória pêndulo com haste elástica e dissipação

```

Clear[l, m, Ke, Kd, t];
l = 0.5; m = 0.5; Kd = 0.4; Ke = 15; g = -9.81;
s = NDSolve[{
  (1 + \[Delta][t]) ((Kd l + Kd \[Delta][t] + 2 m \[Delta]'[t])
    \[Theta]'[t] + m (g Sin\[Theta][t] + (1 + \[Delta][t])
    \[Theta]''[t])) == 0,
  Kd \[Delta]'[t] + \[Delta][t] (Ke - m \[Theta]'[t]^2) +
    m (-g Cos\[Theta][t] - 1 \[Theta]'[t]^2 + \[Delta]''[t])
    == 0
, \[Theta][0] == 2.57
, \[Theta]'[0] == 1
, \[Delta][0] == 0.2
, \[Delta]'[0] == 0.1
}
, {\[Theta], \[Delta]}
, {t, 0, 100}
];

```

```

inicio = 0;
fim = 5;
PontoInicial = Evaluate[{(1 + \[Delta][inicio]) Sin\[Theta][inicio]
, (1 + \[Delta][inicio]) Cos\[Theta][inicio]] /. s];
PontoFinal = Evaluate[{(1 + \[Delta][fim]) Sin\[Theta][fim]
, (1 + \[Delta][fim]) Cos\[Theta][fim]] /. s];
Legendas = PointLegend[{Orange, Purple}
, {"Ponto Inicial ", "Ponto Final "});
G1 = ParametricPlot[Evaluate[
{(1 + \[Delta][t]) Sin\[Theta][t]
, (1 + \[Delta][t]) Cos\[Theta][t]] /. s]
, {t, inicio, fim}
, PlotStyle -> Thick, PlotRange -> {{-0.5, 0.5}, {-1, -0.5}}
, Axes -> {True, True}, AxesOrigin -> {0, 0}
, PlotLegends -> Legendas
];
G2 = Graphics[{Orange, Disk[PontoInicial[[1]], 0.01]}];
G3 = Graphics[{Purple, Disk[PontoFinal[[1]], 0.01]}];
Show[G1, G2, G3]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.13: Pêndulo plano com haste elástica e dissipação

```

Clear[l, m, Ke, Kd, t, te]
g = -9.81;
Manipulate[
s = NDSolve[{
(1 + \[Delta][te]) ((Kd 1 + Kd \[Delta][te] + 2 m \[Delta]'[te])
\[Theta]'[te] + m (g Sin\[Theta][te] +
(1 + \[Delta][te]) \[Theta]''[te])) == 0,
Kd \[Delta]'[te] + \[Delta][te] (Ke - m \[Theta]'[te]^2) +
m (-g Cos\[Theta][te] - 1 \[Theta]'[te]^2 + \[Delta]''[te])
== 0
, \[Theta]'[0] == d\[Theta]0
, \[Theta][0] == \[Theta]0
, \[Delta]'[0] == d\[Delta]0
, \[Delta][0] == \[Delta]0
}
, {\[Theta], \[Delta]}
, {te, Max[-1, t - 30], t + 1}
];
Show[
ParametricPlot[
Evaluate[{(1 + \[Delta][i]) Sin\[Theta][i],
(1 + \[Delta][i]) Cos\[Theta][i]] /. s]
, {i, Max[0.01, t - r], t}
, PlotStyle -> {Thick}
, PlotRange -> 20

```

```

, Axes -> {True, True}, AxesOrigin -> {0, 0}
, ImageSize -> Large
]
, Graphics[{Thick, Black,
  Line[{0, 0},
    Evaluate[{(1 + \[Delta][t]) Sin\[Theta][t], (1 + \[Delta][t])
      Cos\[Theta][t]} /. s][[1]]},
  Disk[Evaluate[{(1 + \[Delta][t]) Sin\[Theta][t],
    (1 + \[Delta][t]) Cos\[Theta][t]} /. s][[1]],
    m 0.15]
  }
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
  AppearanceElements -> {"PlayPauseButton", "ResetButton",
    "FasterSlowerButtons", "ProgressSlider"}, Trigger,
  Appearance -> "Labeled", AnimationRate -> 2}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. haste"}, 0.5, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Ke, 1.5, "Cte. elastica"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Kd, 0.3, "Cte. dissipacao"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Theta]0, 0, "Vel. ang. \[Theta] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Theta]0, 1.57, "\[Theta] inicial"}, -3.14, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d\[Delta]0, 0, "Vel. \[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Delta]0, 0, "\[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 7, "rastros"}, 0.01, 30, Slider, ImageSize -> Tiny}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.14: Trajetória pêndulo esférico

```
Clear[m, g, l, t]
```

```

m = 0.7; g = -9.81; l = 2;
s = NDSolve[{
  2 \[Theta]'[t] \[Phi]'[t] Cos\[Phi][t] Sin\[Phi][t]
  + \[Theta]''[t] Sin\[Phi][t]^2 == 0,
  g Sin\[Phi][t] - l \[Theta]'[t]^2 Cos\[Phi][t]
  Sin\[Phi][t] + l \[Phi]''[t] == 0
  , \[Theta][0] == \[Pi]
  , \[Theta]'[0] == 1
  , \[Phi][0] == 1.6
  , \[Phi]'[0] == 0.1
}
, {\[Theta], \[Phi]}
, {t, 0, 100}
];
inicio = 0;
fim = 5.5;
PontoInicial = Evaluate[
  {l Cos\[Theta][inicio] Sin\[Phi][inicio]
  , l Sin\[Theta][inicio] Sin\[Phi][inicio]
  , l Cos\[Phi][inicio]} /. s];
PontoFinal = Evaluate[
  {l Cos\[Theta][fim] Sin\[Phi][fim]
  , l Sin\[Theta][fim] Sin\[Phi][fim]
  , l Cos\[Phi][fim]} /. s];
Legendas = PointLegend[{Orange, Purple}
  , {"Ponto Inicial", "Ponto Final"}];
G1 = ParametricPlot3D[
  {l Cos\[Theta][t] Sin\[Phi][t]
  , l Sin\[Theta][t] Sin\[Phi][t]
  , l Cos\[Phi][t]} /. s
  , {t, inicio, fim}
  , PlotLegends -> Legendas
];
G2 = Graphics3D[{Orange, Sphere[PontoInicial[[1]], 0.05]}];
G3 = Graphics3D[{Purple, Sphere[PontoFinal[[1]], 0.05]}];
Show[G1, G2, G3]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.15: Pêndulo esférico

```

Clear[l, m, t, te]
g = -9.81;
Manipulate[
  s = NDSolve[{
    2 \[Theta]'[te] \[Phi]'[te] Cos\[Phi][te] Sin\[Phi][te] +
    \[Theta]''[te] Sin\[Phi][te]^2 == 0,
    g Sin\[Phi][te] - l \[Theta]'[te]^2 Cos\[Phi][te]
    Sin\[Phi][te] + l \[Phi]''[te] == 0
  }

```

```

, \[Theta]'[0] == d\[Theta]0
, \[Theta][0] == \[Theta]0
, \[Phi]'[0] == d\[Phi]0
, \[Phi][0] == \[Phi]0
}
, {\[Theta], \[Phi]}
, {te, Max[-1, t - 30], t + 1}
];
Show[
Graphics3D[{Thick, Black,
  Line[{0, 0, 0},
    Evaluate[{1 Cos\[Theta][t] Sin\[Phi][t],
      1 Sin\[Theta][t] Sin\[Phi][t]
      , 1 Cos\[Phi][t]} /. s][[1]]
    ]],
  Sphere[
    Evaluate[{1 Cos\[Theta][t] Sin\[Phi][t],
      1 Sin\[Theta][t] Sin\[Phi][t]
      , 1 Cos\[Phi][t]} /. s][[1]]
    , m 0.15
  ]
}
, PlotRange -> 6.3
, Axes -> {True, True, True}, AxesOrigin -> {0, 0, 0}
, ImageSize -> Large
, ViewPoint -> vp
]
, ParametricPlot3D[
  Evaluate[{1 Cos\[Theta][i] Sin\[Phi][i],
    1 Sin\[Theta][i] Sin\[Phi][i], 1 Cos\[Phi][i]} /. s]
  , {i, Max[0.01, t - r], t}, PlotStyle -> {Thick}
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
  AppearanceElements -> {"PlayPauseButton", "ResetButton",
    "FasterSlowerButtons", "ProgressSlider"}, Trigger,
  Appearance -> "Labeled", AnimationRate -> 1}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. haste"}, 0.5, 6, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Theta]0, 0.3, "Vel. ang. \[Theta] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}

```

```

, {{\[Theta]0, 1.57, "\[Theta] inicial"}, -3.14, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d\[Phi]0, 0, "Vel. ang. \[Phi] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Phi]0, 1, "\[Phi] inicial"}, 0.01, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 7, "rastro"}, 0.01, 30, Slider, ImageSize -> Tiny}
, {{vp, {1.2, -1, 1.25},
  "Ponto de vista"}, {{1.2, -1, 1.25} -> "padrao", Front -> "frente",
  Top -> "topo"}, SetterBar}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.16: Trajetória pêndulo esférico com dissipação

```

Clear[l, m, t, t]
m = 0.7; l = 2; Kd = 0.5; g = -9.81;
s = NDSolve[{
  l (g m Sin\[Phi][t] - l m Cos\[Phi][t] Sin\[Phi][t]
    \[Theta]'[t]^2 + Kd l \[Phi]'[t] + l m \[Phi]''[t]) == 0,
  l^2 Sin\[Phi][t] (\[Theta]'[t] (Kd Sin\[Phi][t]
    + 2 m Cos\[Phi][t] \[Phi]'[t]) + m Sin\[Phi][t]
    \[Theta]''[t]) == 0
  , \[Theta][0] == 0
  , \[Theta]'[0] == -1
  , \[Phi][0] == 1.57
  , \[Phi]'[0] == 0.1
}
, {\[Theta], \[Phi]}
, {t, 0, 30}
];
inicio = 0;
fim = 5.5;
PontoInicial = Evaluate[
  {l Cos\[Theta][inicio] Sin\[Phi][inicio]
  , l Sin\[Theta][inicio] Sin\[Phi][inicio]
  , l Cos\[Phi][inicio]] /. s];
PontoFinal = Evaluate[
  {l Cos\[Theta][fim] Sin\[Phi][fim]
  , l Sin\[Theta][fim] Sin\[Phi][fim]
  , l Cos\[Phi][fim]] /. s];
Legendas = PointLegend[{Orange, Purple}
  , {"Ponto Inicial", "Ponto Final"}];
G1 = ParametricPlot3D[
  Evaluate[

```

```

    {1 Cos[\[Theta][t]] Sin\[Phi][t],
      1 Sin\[Theta][t] Sin\[Phi][t], 1 Cos\[Phi][t]} /. s
  ]
, {t, inicio, fim}
, PlotStyle -> {Thick}
, PlotRange -> 2.1
, PlotLegends -> Legendas
];
G2 = Graphics3D[{Orange, Sphere[PontoInicial[[1]], 0.06]}];
G3 = Graphics3D[{Purple, Sphere[PontoFinal[[1]], 0.06]}];
Show[G1, G2, G3]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.17: Pêndulo esférico com dissipação

```

Clear[l, m, t, te];
g = -9.81;
Manipulate[
  s = NDSolve[{
    l (g m Sin\[Phi][te] -
      1 m Cos\[Phi][te] Sin\[Phi][te] \[Theta]'[te]^2 +
      Kd 1 \[Phi]'[te] + 1 m \[Phi]''[te]) == 0,
    l^2 Sin\[Phi][te] (\[Theta]'[te] (Kd Sin\[Phi][te] +
      2 m Cos\[Phi][te] \[Phi]'[te]) +
      m Sin\[Phi][te] \[Theta]''[te]) == 0
  }, \[Theta]'[0] == d\[Theta]0
  , \[Theta][0] == \[Theta]0
  , \[Phi]'[0] == d\[Phi]0
  , \[Phi][0] == \[Phi]0
  }
  , {\[Theta], \[Phi]}
  , {te, Max[-1, t - 30], t + 1}
];
Show[
  Graphics3D[{Thick, Black,
    Line[{0, 0, 0},
      Evaluate[{1 Cos\[Theta][t] Sin\[Phi][t],
        1 Sin\[Theta][t] Sin\[Phi][t],
        1 Cos\[Phi][t]} /. s][[1]]}],
    Sphere[
      Evaluate[{1 Cos\[Theta][t] Sin\[Phi][t],
        1 Sin\[Theta][t] Sin\[Phi][t],
        1 Cos\[Phi][t]} /. s][[1]], m 0.15]
  ]
  , PlotRange -> 6.3
  , Axes -> {True, True, True}, AxesOrigin -> {0, 0, 0}
  , ImageSize -> Large
  , ViewPoint -> vp

```



```

]
, ParametricPlot3D[
  Evaluate[{1 Cos\[Theta][i] Sin\[Phi][i],
    1 Sin\[Theta][i] Sin\[Phi][i], 1 Cos\[Phi][i]} /. s]
  , {i, Max[0.01, t - r], t}
  , PlotStyle -> {Thick}
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
  AppearanceElements -> {"PlayPauseButton", "ResetButton",
    "FasterSlowerButtons", "ProgressSlider"}, Trigger,
  Appearance -> "Labeled", AnimationRate -> 1}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. haste"}, 0.5, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Kd, 0.3, "Cte. dissipacao"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Theta]0, 1, "Vel. ang. \[Theta] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Theta]0, 0, "\[Theta] inicial"}, -3.14, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d\[Phi]0, 0, "Vel. ang. \[Phi] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Phi]0, 1.57, "\[Phi] inicial"}, 0.01, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 10, "rastro"}, 0.01, 30, Slider, ImageSize -> Tiny}
, {{vp, {1.2, -1, 1.25},
  "Ponto de vista"}, {{1.2, -1, 1.25} -> "padrao", Front -> "frente",
  Top -> "topo"}, SetterBar}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.18: Trajetória pêndulo esférico com haste elástica

```

Clear[m, l, Ke, g];
m = 1; l = 2; Ke = 6; g = -9.81;
s = NDSolve[
{
  m Sin\[Phi][t] (1 + \[Delta][t]) (2 Sin\[Phi][t] \[Delta]'[t]

```

```

\[\Theta]'[t] + (1 + \[\Delta][t]) (2 Cos[\[\Phi][t]]
\[\Theta]'[t] \[\Phi]'[t] + Sin[\[\Phi][t]]
\[\Theta]''[t])) == 0,
m (1 + \[\Delta][t]) (-Sin[\[\Phi][t]] (-g + Cos[\[\Phi][t]]
(1 + \[\Delta][t]) \[\Theta]'[t]^2) + 2 \[\Delta]'[t]
\[\Phi]'[t] + (1 + \[\Delta][t]) \[\Phi]''[t]) == 0,
-m (g Cos[\[\Phi][t]] + 1 Sin[\[\Phi][t]]^2 \[\Theta]'[t]^2
+ 1 \[\Phi]'[t]^2) + \[\Delta][t] (Ke - m Sin[\[\Phi][t]]^2
\[\Theta]'[t]^2 - m \[\Phi]'[t]^2) + m \[\Delta]''[t] == 0
, \[\Theta][0] == 0
, \[\Theta]'[0] == 1
, \[\Phi][0] == 1.57
, \[\Phi]'[0] == -0.1
, \[\Delta][0] == -1
, \[\Delta]'[0] == 0.3
}
, { \[\Theta], \[\Phi], \[\Delta]}
, {t, 0, 30}
];
inicio = 0;
fim = 7;
PontoInicial = Evaluate[
{(1 + \[\Delta][inicio]) Sin[\[\Phi][inicio]] Cos[\[\Theta][inicio]]
, (1 + \[\Delta][inicio]) Sin[\[\Phi][inicio]] Sin[\[\Theta][inicio]]
, (1 + \[\Delta][inicio]) Cos[\[\Phi][inicio]]} /. s];
PontoFinal = Evaluate[
{(1 + \[\Delta][fim]) Sin[\[\Phi][fim]] Cos[\[\Theta][fim]]
, (1 + \[\Delta][fim]) Sin[\[\Phi][fim]] Sin[\[\Theta][fim]]
, (1 + \[\Delta][fim]) Cos[\[\Phi][fim]]} /. s];
Legendas = PointLegend[{Orange, Purple}
, {"Ponto Inicial", "Ponto Final"}];
G1 = ParametricPlot3D[
Evaluate[
{(1 + \[\Delta][t]) Sin[\[\Phi][t]] Cos[\[\Theta][t]]
, (1 + \[\Delta][t]) Sin[\[\Phi][t]] Sin[\[\Theta][t]]
, (1 + \[\Delta][t]) Cos[\[\Phi][t]]} /. s
]
, {t, inicio, fim}
, PlotRange -> All
, PlotLegends -> Legendas
];
G2 = Graphics3D[{Orange, Sphere[PontoInicial[[1]], 0.1]}];
G3 = Graphics3D[{Purple, Sphere[PontoFinal[[1]], 0.1]}];
Show[G1, G2, G3]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

```

Clear[l, m, t, te, Ke];
g = -9.81;
Manipulate[
  s = NDSolve[{
    m Sin[\[Phi][te] (1 + \[Delta][te]) (2 Sin[\[Phi][te]
      \[Delta]'[te] \[Theta]'[te] + (1 + \[Delta][te])
      (2 Cos[\[Phi][te] \[Theta]'[te] \[Phi]'[te] + Sin[\[Phi][te]
      \[Theta]''[te])) == 0,
    m (1 + \[Delta][te]) (-Sin[\[Phi][te] (-g + Cos[\[Phi][te]
      (1 + \[Delta][te]) \[Theta]'[te]^2) + 2 \[Delta]'[te] \[Phi]'[te] +
      (1 + \[Delta][te]) \[Phi]''[te]) == 0,
    -m (g Cos[\[Phi][te] + 1 Sin[\[Phi][te]^2 \[Theta]'[te]^2 +
      1 \[Phi]'[te]^2) + \[Delta][te] (Ke - m Sin[\[Phi][te]^2
      \[Theta]'[te]^2 - m \[Phi]'[te]^2) + m \[Delta]''[te] == 0
    , \[Theta]'[0] == d\[Theta]0
    , \[Theta][0] == \[Theta]0
    , \[Phi]'[0] == d\[Phi]0
    , \[Phi][0] == \[Phi]0
    , \[Delta]'[0] == d\[Delta]0
    , \[Delta][0] == \[Delta]0
  }
  , {\[Theta], \[Phi], \[Delta]}
  , {te, Max[-1, t - 30], t + 1}
];
Show[
  Graphics3D[{Thick, Black,
    Line[{0, 0, 0}, Evaluate[{(1 + \[Delta][t]) Sin[\[Phi][t]]
      Cos[\[Theta][t]], (1 + \[Delta][t]) Sin[\[Phi][t]]
      Sin[\[Theta][t]], (1 + \[Delta][t]) Cos[\[Phi][t]]} /. s][[1]]}],
    Sphere[
      Evaluate[{(1 + \[Delta][t]) Sin[\[Phi][t]) Cos[\[Theta][t]],
        (1 + \[Delta][t]) Sin[\[Phi][t]) Sin[\[Theta][t]],
        (1 + \[Delta][t]) Cos[\[Phi][t]]} /. s][[1]],
      m 0.15]
    ]
  , PlotRange -> 20
  , Axes -> {True, True, True}, AxesOrigin -> {0, 0, 0}
  , ImageSize -> Large
  , ViewPoint -> vp
]
, ParametricPlot3D[
  Evaluate[{(1 + \[Delta][i]) Sin[\[Phi][i]) Cos[\[Theta][i]],
    (1 + \[Delta][i]) Sin[\[Phi][i]) Sin[\[Theta][i]],
    (1 + \[Delta][i]) Cos[\[Phi][i]]} /. s]
  , {i, Max[0.01, t - r], t}, PlotStyle -> {Thick}]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,

```

```

AppearanceElements -> {"PlayPauseButton", "ResetButton",
  "FasterSlowerButtons", "ProgressSlider"}, Trigger,
Appearance -> "Labeled", AnimationRate -> 1}
, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. rep. mola"}, 0.5, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Ke, 4, "Cte. elastica"}, 0, 10, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d\[Theta]0, 0.5, "Vel. ang. \[Theta] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Theta]0, 0, "\[Theta] inicial"}, -3.14, 4.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d\[Phi]0, 0.5, "Vel. ang. \[Phi] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Phi]0, 1.57, "\[Phi] inicial"}, 0.01, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d\[Delta]0, 0, "Vel. \[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Delta]0, 1, "\[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 20, "rastro"}, 0.01, 30, Slider, ImageSize -> Tiny}
, {{vp, {1.2, -1, 1.25}, "Ponto de vista"}
  , {{1.2, -1, 1.25} -> "padrao",
  Front -> "frente", Top -> "topo"}, SetterBar}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.20: Trajetória pêndulo esférico com haste elástica e dissipação

```

Clear[m, g, l, Kd, Ke, t]
m = 1; g = -9.81; l = 1; Kd = 0.5; Ke = 10;
s = NDSolve[{
  Sin\[Phi][t] (1 + \[Delta][t]) (\[Theta]'[t] (Kd l Sin\[Phi][t]
    + 2 m Sin\[Phi][t] \[Delta]'[t] + 2 l m Cos\[Phi][t] \[Phi]'[t])
    + \[Delta][t] (Kd Sin\[Phi][t] + 2 m Cos\[Phi][t] \[Phi]'[t]))
    + m Sin\[Phi][t] (1 + \[Delta][t]) \[Theta]''[t]) == 0,
  (1 + \[Delta][t]) (-m Cos\[Phi][t] Sin\[Phi][t]
    (1 + \[Delta][t]) \[Theta]'[t]^2 + (Kd l + Kd \[Delta][t]
    + 2 m \[Delta]'[t]) \[Phi]'[t] + m (g Sin\[Phi][t]
    + (1 + \[Delta][t]) \[Phi]''[t])) == 0,

```

```

Kd \[Delta]'[t] + \[Delta][t] (Ke - m Sin\[Phi][t]^2 \[Theta]'[t]^2
- m \[Phi]'[t]^2) -m (g Cos\[Phi][t] + l Sin\[Phi][t]^2
\[Theta]'[t]^2 + l \[Phi]'[t]^2 - \[Delta]''[t]) == 0,
\[Theta][0] == 0,
\[Theta]'[0] == 1,
\[Phi][0] == 1.57,
\[Phi]'[0] == 0.1,
\[Delta][0] == 0.5,
\[Delta]'[0] == -0.4
}
, {\[Theta], \[Phi], \[Delta]}
, {t, 0, 100}
];
inicio = 0;
fim = 5;
PontoInicial = Evaluate[
{(1 + \[Delta][inicio]) Sin\[Phi][inicio] Cos\[Theta][inicio]
, (1 + \[Delta][inicio]) Sin\[Phi][inicio] Sin\[Theta][inicio]
, (1 + \[Delta][inicio]) Cos\[Phi][inicio]] /. s};
PontoFinal = Evaluate[
{(1 + \[Delta][fim]) Sin\[Phi][fim] Cos\[Theta][fim]
, (1 + \[Delta][fim]) Sin\[Phi][fim] Sin\[Theta][fim]
, (1 + \[Delta][fim]) Cos\[Phi][fim]] /. s};
Legendas = PointLegend[{Orange, Purple}
, {"Ponto Inicial", "Ponto Final"}];
G1 = ParametricPlot3D[
Evaluate[{{(1 + \[Delta][t]) Sin\[Phi][t] Cos\[Theta][t]
, (1 + \[Delta][t]) Sin\[Phi][t] Sin\[Theta][t]
, (1 + \[Delta][t]) Cos\[Phi][t]] /. s
}
, {t, inicio, fim}
, PlotStyle -> {Thick}
, PlotRange -> 3.2
, PlotLegends -> Legendas
];
G2 = Graphics3D[{Orange, Sphere[PontoInicial[[1]], 0.1]}];
G3 = Graphics3D[{Purple, Sphere[PontoFinal[[1]], 0.1]}];
Show[G1, G2, G3]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.

Código A.21: Pêndulo esférico com haste elástica e dissipação

```

Clear[l, m, t, te, Ke, Kd];
g = -9.81;
Manipulate[
s = NDSolve[{Sin\[Phi][te] (1 + \[Delta][te]) (\[Theta]'[te]
(Kd l Sin\[Phi][te] + 2 m Sin\[Phi][te] \[Delta]'[te]
+ 2 l m Cos\[Phi][te] \[Phi]'[te] + \[Delta][te]

```

```

(Kd Sin[\[Phi][te]] + 2 m Cos[\[Phi][te]] \[Phi]'[te]))
+ m Sin[\[Phi][te]] (1 + \[Delta][te]) \[Theta]''[te]) == 0,
(1 + \[Delta][te]) (-m Cos[\[Phi][te]] Sin[\[Phi][te]]
(1 + \[Delta][te]) \[Theta]'[te]^2 + (Kd 1 + Kd \[Delta][te] +
2 m \[Delta]'[te]) \[Phi]'[te] + m (g Sin[\[Phi][te]] +
(1 + \[Delta][te]) \[Phi]''[te])) == 0,
Kd \[Delta]'[te] + \[Delta][te] (Ke - m Sin[\[Phi][te]]^2
\[Theta]'[te]^2 - m \[Phi]'[te]^2) - m (g Cos[\[Phi][te]] +
1 Sin[\[Phi][te]]^2 \[Theta]'[te]^2 + 1 \[Phi]'[te]^2 -
\[Delta]''[te]) == 0
, \[Theta]'[0] == d\[Theta]0
, \[Theta][0] == \[Theta]0
, \[Phi]'[0] == d\[Phi]0
, \[Phi][0] == \[Phi]0
, \[Delta]'[0] == d\[Delta]0
, \[Delta][0] == \[Delta]0
}
, {\[Theta], \[Phi], \[Delta]}
, {te, Max[-1, t - 30], t + 1}
];
Show[
Graphics3D[{Thick, Black,
Line[{{0, 0, 0},
Evaluate[{{(1 + \[Delta][t]) Sin[\[Phi][t]] Cos[\[Theta][t]],
(1 + \[Delta][t]) Sin[\[Phi][t]] Sin[\[Theta][t]],
(1 + \[Delta][t]) Cos[\[Phi][t]]} /. s[[1]]}],
Sphere[
Evaluate[{{(1 + \[Delta][t]) Sin[\[Phi][t]] Cos[\[Theta][t]],
(1 + \[Delta][t]) Sin[\[Phi][t]] Sin[\[Theta][t]],
(1 + \[Delta][t]) Cos[\[Phi][t]]} /. s[[1]], m 0.15]
}
, PlotRange -> 20
, Axes -> {True, True, True}, AxesOrigin -> {0, 0, 0}
, ImageSize -> Large
]
, ParametricPlot3D[
Evaluate[{{(1 + \[Delta][i]) Sin[\[Phi][i]] Cos[\[Theta][i]],
(1 + \[Delta][i]) Sin[\[Phi][i]] Sin[\[Theta][i]],
(1 + \[Delta][i]) Cos[\[Phi][i]]} /. s
, {i, Max[0.01, t - r], t}
, PlotStyle -> {Thick}
]
]
, Style["Controle animacao", Bold]
, {{t, 0, "Tempo"}, 0, 100, ImageSize -> Tiny,
AppearanceElements -> {"PlayPauseButton", "ResetButton",
"FasterSlowerButtons", "ProgressSlider"}, Trigger,
Appearance -> "Labeled", AnimationRate -> 1}

```

```

, Delimiter
, Style["Parametros do sistema", Bold]
, {{m, 1, "Massa"}, 0.1, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{l, 1, "Comp. haste"}, 0.5, 3, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Ke, 1.5, "Cte. elastica"}, 0, 6, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, {{Kd, 0.3, "Cte. dissipacao"}, 0, 6, Slider, ImageSize -> Tiny,
  Appearance -> "Labeled"}
, Delimiter
, Style["Condicoes iniciais", Bold]
, {{d[Theta]0, 0.3, "Vel. ang. \[Theta] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Theta]0, 0, "\[Theta] inicial"}, -3.14, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d[Phi]0, 0, "Vel. ang. \[Phi] inicial"}, -10, 10, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Phi]0, 1.57, "\[Phi] inicial"}, 0.01, 3.14, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{d[Delta]0, 0, "Vel. \[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, {{\[Delta]0, 1, "\[Delta] inicial"}, -2, 2, Slider,
  ImageSize -> Tiny, Appearance -> "Labeled"}
, Delimiter
, Style["Seguir", Bold]
, {{r, 7, "rastro"}, 0.01, 30, Slider, ImageSize -> Tiny}
, ControlPlacement -> Left
]

```

Código disponível para *download* nos formatos **cdf** ou **nb**.